INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 0000; **00**:1–26
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme

Fluid Structure Interaction by means of Variational Multiscale Reduced Order Models

Alexis Tello[1], Ramon Codina[1,2], and Joan Baiges[1,*]

[1] *Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08034 Barcelona, Spain*
[2] *International Centre for Numerical Methods in Engineering (CIMNE), C/ Gran Capità S/N, 08034 Barcelona, Spain*

SUMMARY

A Reduced Order Model (ROM) designed by means of a Variational Multi-Scale (VMS) stabilized formulation has been applied successfully to Fluid-Structure Interaction (FSI) problems in a strongly coupled partitioned solution scheme. Details of the formulation and the implementation both for the interaction problem and for the reduced models, for both the off-line and on-line phases, are shown. Results are obtained for cases in which both domains are reduced at the same time. Numerical results are presented for a semi-stationary and a fully transient case. Copyright © 0000 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: Fluid Structure Interaction (FSI); Reduced Order Model (ROM); Variational Multi-Scale Method (VMS); Non-linear Solid Elasto-dynamics

1. INTRODUCTION

Fluid Structure Interaction (FSI) is a topic of constant research and development, and even though fluid and solid formulations might be well understood, FSI remains a complex problem owing to factors such as the added mass effect, instabilities of the fluid and solid problems, and the overall conditioning of the problem. Broadly, research in the field can be grouped into two categories based on how the mesh is treated, namely, conforming and non conforming methods. Essentially, conforming mesh methods consider interface conditions as physical boundary conditions, thus treating the interface as part of the solution. In this approach, the mesh reproduces or conforms to the interface; when the interface is moved it is also necessary to displace the mesh, which carries on all related problems of mesh recalculation and inherent instabilities of the method, be it partitioned or monolithic, see [2, 22, 23, 12, 10, 9, 32]. On the other hand, non-conforming methods treat the interface and boundary as constraints imposed on the governing equations, which makes possible to use meshes that do not reproduce the interface; the main problem in this case is the treatment of the interface conditions and the complexity of the formulation, see for example [7, 26] for further reading. For a general review of significant FSI advances and developments, see [27]. Overall, for highly non-linear problems, arriving at a solution can take a large amount of time, an issue that becomes even more apparent when dealing with problems with a high number of degrees of freedom. It is well known that Reduced Order Models (ROM) can speed up solution time dramatically, which leads to the idea of introducing them into FSI analysis.

*Correspondence to: Corresponding author joan.baiges@upc.edu (J. Baiges)

Model Order Reduction (MOR) was originally developed for the area of system control theory, its main purpose being reducing its complexity while maintaining the input-output behavior. The resulting mathematical approximation to the original full order problem is precisely known as a Reduced Order Model. From this, MOR rapidly spread to other fields of research quite successfully. Various ways to achieve model reduction and achieve solution speed up are available, see [4, 5, 35, 36, 20]. Amongst the various families of reduced order models, Proper Orthogonal Decomposition (POD) gained considerable attention in the area of numerical analysis, particularly in fluid dynamics, because of its applicability to non linear partial differential equations. POD is the foundation of the methods used in this work.

In terms of recent FSI-ROM work, we can refer to [18], where the fluid domain hyper-reduction is obtained by means of POD-Greedy algorithms applied to the field of haemodynamics. In [8] the authors propose a POD approach for a monolithic FSI where the basis for the resulting system is obtained in a monolithic way, both for the Newtonian fluid and the linear elastic solid. The idea of their method is the parametrization of variables by means of empirical interpolations, providing accurate results for a range of data considered in the interpolation charts. In [39] the authors introduce the concept of non intrusive model reduction to the FSI field, making the calculation of the reduced basis problem independently for the fluid and the solid. In terms of solid domain reduction, in [37] the authors apply a modal analysis by means of model recalibration to the movement of a membrane.

Our approach is different, we consider the reduced system to be variational by nature and use the need of stabilization as a way to project the solution of both the fluid and the solid onto the reduced space. This is not the first time this is done however; see for example [6], where the idea of *sub-grid scales* for ROM is first explored, [25], where for the first time stabilization trough the residual of the reduced space is considered, and finally [33], where the residual of the reduced problem is projected into the reduced space where the solution is taken. We apply this formulation in this work to FSI problems.

The paper is organized as follows: in Sections 2 and 3 we describe each particular formulation, the fluid and solid, respectively, in a short manner. In Section 4 we introduce the FSI formulation. Afterwards, an overview of ROM will be given in Section 5, detailing our implementation. Numerical results are presented in Section 6 and finally conclusions close the paper in Section 7.

2. INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

In the next section we present the finite element approximation we employ to solve the incompressible Navier-Stokes equations. This approximation is what we call the Full Order Model (FOM) for the fluid problem.

2.1. Governing equations

Let Ω_{fl} be the domain where the fluid flow takes place, with boundary $\Gamma_{\mathrm{fl}} = \Gamma_{\mathrm{D,fl}} \cup \Gamma_{\mathrm{N,fl}}$, where $\Gamma_{\mathrm{D,fl}}$ and $\Gamma_{\mathrm{N,fl}}$ are boundaries where Dirichlet and Neumann conditions are prescribed, respectively. Let $[0, t_{\mathrm{f}}]$ be the time interval of analysis. The incompressible Navier-Stokes equations can be written as finding a velocity-pressure pair $[\boldsymbol{u}, p] : \Omega_{\mathrm{fl}} \times [0, t_{\mathrm{f}}[\longrightarrow \mathbb{R}^d \times \mathbb{R}$, where d is the space dimension, as the solution to the following equations:

$$
\begin{aligned}
\rho_{\mathrm{fl}} \partial_t \boldsymbol{u} - 2\mu_{\mathrm{fl}} \nabla \cdot \nabla^s \boldsymbol{u} + \rho_{\mathrm{fl}} \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p &= \rho_{\mathrm{fl}} \boldsymbol{f} && \text{in } \Omega_{\mathrm{fl}}, && t \in]0, t_{\mathrm{f}}[, \\
\nabla \cdot \boldsymbol{u} &= 0 && \text{in } \Omega_{\mathrm{fl}}, && t \in]0, t_{\mathrm{f}}[, \\
\boldsymbol{u} &= \boldsymbol{u}_{\mathrm{D}} && \text{on } \Gamma_{\mathrm{D,fl}}, && t \in]0, t_{\mathrm{f}}[,
\end{aligned}
$$

DOI: 10.1002/nme

$$\begin{aligned} \boldsymbol{n}_{\mathrm{fl}} \cdot \boldsymbol{\sigma}_{\mathrm{fl}} &= \boldsymbol{t}_{\mathrm{fl}} && \text{on } \Gamma_{\mathrm{N,fl}}, && t \in]0, t_{\mathrm{f}}[, \\ \boldsymbol{u} &= \boldsymbol{u}^0 && \text{in } \Omega_{\mathrm{fl}}, && t = 0, \end{aligned}$$

where ρ_{fl} is the fluid's density, μ_{fl} the fluid's dynamic viscosity, $\nabla^s \boldsymbol{u}$ the symmetrical part of the velocity gradient, $\boldsymbol{f}$ the body acceleration vector, $\boldsymbol{\sigma}_{\mathrm{fl}} = -p\boldsymbol{I} + 2\mu_{\mathrm{fl}}\nabla^s \boldsymbol{u}$ the fluid's Cauchy stress tensor ($\boldsymbol{I}$ being the identity tensor), $\boldsymbol{u}^0$ a prescribed initial velocity, $\boldsymbol{u}_{\mathrm{D}}$ a prescribed velocity on the boundary $\Gamma_{\mathrm{D,fl}}$, $\boldsymbol{t}_{\mathrm{fl}}$ is a prescribed traction on the boundary $\Gamma_{\mathrm{N,fl}}$, and $\boldsymbol{n}_{\mathrm{fl}}$ the normal to the boundary.

2.2. Weak form

Let us introduce some standard notation. The space of functions whose p power ($p \geq 1$) is integrable in a domain Ω is denoted by $L^p(\Omega)$, and the space of functions whose distributional derivatives of order up to $m \geq 0$ belong to $L^2(\Omega)$ by $H^m(\Omega)$. The L^2 inner product in Ω (for scalars, vectors or tensors) is denoted by $(\cdot, \cdot)$. The integral of the product of two functions defined in ω is $\langle \cdot, \cdot \rangle_\omega$, with the subscript omitted when $\omega = \Omega$; this definition includes the duality pairing. Given a Banach space X of time dependent functions, $L^p(0, t_{\mathrm{f}}; X)$ denotes the space of functions whose norm in X is in $L^p(0, t_{\mathrm{f}})$, $p \geq 1$.

Using this notation we can introduce the spaces for the Navier-Stokes equations, where now $\Omega = \Omega_{\mathrm{fl}}$. Let $\mathcal{V}_0 = \{\boldsymbol{v} \in H^1(\Omega_{\mathrm{fl}})^d \mid \boldsymbol{v}|_{\Gamma_D} = \boldsymbol{0}\}$, $\mathcal{V}_D = \{\boldsymbol{v} \in H^1(\Omega_{\mathrm{fl}})^d \mid \boldsymbol{v}|_{\Gamma_D} = \boldsymbol{u}_D\}$, $\mathcal{Q} = L^2(\Omega_{\mathrm{fl}})$, $\mathcal{W}_0 = \mathcal{V}_0 \times \mathcal{Q}$ and $\mathcal{W}_D = \mathcal{V}_D \times \mathcal{Q}$. The weak form of the Navier-Stokes equations consists in finding $[\boldsymbol{u}, p] \in L^2(0, t_{\mathrm{f}}; \mathcal{V}_D) \times L^1(0, t_{\mathrm{f}}; \mathcal{Q})$ (or a distribution in time) such that:

$$(\rho_{\mathrm{fl}} \partial_t \boldsymbol{u}, \boldsymbol{v}) - 2\mu_{\mathrm{fl}}(\nabla^s \boldsymbol{u}, \nabla^s \boldsymbol{v}) + \rho_{\mathrm{fl}} \langle \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \rangle - (p, \nabla \cdot \boldsymbol{v}) = \langle \rho_{\mathrm{fl}} \boldsymbol{f}, \boldsymbol{v} \rangle + \langle \boldsymbol{t}, \boldsymbol{v} \rangle_{\Gamma_{N,\mathrm{fl}}}, \quad t \in]0, t_{\mathrm{f}}[, \tag{1}$$

$$(q, \nabla \cdot \boldsymbol{u}) = 0, \qquad t \in]0, t_{\mathrm{f}}[, \tag{2}$$

$$(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u}^0, \boldsymbol{v}), \qquad t = 0, \tag{3}$$

for all $[\boldsymbol{v}, q] \in \mathcal{V}_0 \times \mathcal{Q}$. For $\boldsymbol{U} \equiv [\boldsymbol{u}, p] :]0, t_{\mathrm{f}}[\longrightarrow \mathcal{W}_D$ and $\boldsymbol{V} \equiv [\boldsymbol{v}, q] \in \mathcal{W}_0$, we can define the form B as

$$B(\boldsymbol{U}, \boldsymbol{V}) = 2\mu_{\mathrm{fl}}(\nabla^s \boldsymbol{u}, \nabla^s \boldsymbol{v}) + \rho_{\mathrm{fl}} \langle \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \rangle - (p, \nabla \cdot \boldsymbol{v}) + (q, \nabla \cdot \boldsymbol{u}),$$

and the linear form L as

$$L(\boldsymbol{V}) = \langle \rho_{\mathrm{fl}} \boldsymbol{f}, \boldsymbol{v} \rangle + \langle \boldsymbol{t}, \boldsymbol{v} \rangle_{\Gamma_{N,\mathrm{fl}}},$$

which enable us to write (1)-(2) in the following simplified form:

$$(\rho_{\mathrm{fl}} \partial_t \boldsymbol{u}, \boldsymbol{v}) + B(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}) \quad \forall\, \boldsymbol{V} \in \mathcal{W}_0. \tag{4}$$

2.3. Galerkin spatial discretization

For the spatial discretization, the standard Galerkin finite element approximation can be defined as follows. Let $\mathcal{P}_h$ denote a finite element partition of a domain Ω. The diameter of an element domain $K \in \mathcal{P}_h$ is denoted by h_K and the diameter of the finite element partition by $h = \max\{h_K | K \in \mathcal{P}_h\}$. In the case $\Omega = \Omega_{\mathrm{fl}}$, we can now construct conforming finite element spaces $\mathcal{V}_h \subset \mathcal{V}_D$, $\mathcal{Q}_h \subset \mathcal{Q}$ and $\mathcal{W}_{h,D} = \mathcal{V}_h \times \mathcal{Q}_h$, as well as $\mathcal{V}_{h,0} \subset \mathcal{V}_0$ and $\mathcal{W}_{h,0} = \mathcal{V}_{h,0} \times \mathcal{Q}_h$, in the usual manner. Then the problem can be written as: find $\boldsymbol{U}_h :]0, t_{\mathrm{f}}[\longrightarrow \mathcal{W}_{h,D}$ as the solution to the problem:

$$\begin{aligned} (\rho_{\mathrm{fl}} \partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + B(\boldsymbol{U}_h, \boldsymbol{V}_h) &= L(\boldsymbol{V}_h) && \forall\, \boldsymbol{V}_h \in \mathcal{W}_{h,0}, \\ (\boldsymbol{u}_h, \boldsymbol{v}_h) &= (\boldsymbol{u}^0, \boldsymbol{v}_h) && \forall\, \boldsymbol{v}_h \in \mathcal{V}_{h,0}, \quad t = 0. \end{aligned} \tag{5}$$

DOI: 10.1002/nme

2.4. Time discretization

Let us consider a uniform partition of the time interval $]0, t_{\mathrm{f}}[$ of size Δt, and let us denote with superscript n the time level. For the temporal discretization, usual finite difference schemes can be adopted. In particular, we have used the second order Backward Difference (BDF2) scheme, which has the following form:

$$\left.\frac{\partial \boldsymbol{u}_h}{\partial t}\right|_{t^{n+1}} = \frac{3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^{n} + \boldsymbol{u}_h^{n-1}}{\Delta t} + \mathcal{O}(\Delta t^2) =: \delta_{2,t}\boldsymbol{u}_h^{n+1} + \mathcal{O}(\Delta t^2),$$

where $t^k = k\Delta t$, $k = n-1, n, n+1$.

2.5. Stabilization

To circumvent the restrictions imposed by the inf-sup condition and convection dominated flows, a Variational Multi-Scale (VMS) stabilization is applied, originally proposed in [29] and later further developed in [14, 16] (see also [17] for a review). When applied to the Navier-Stokes problem, the time discrete version of Eq. (5) is replaced by:

$$(\rho_{\mathrm{fl}}\delta_{2,t}\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) + B(\boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h) + \sum_K \left\langle \tilde{\boldsymbol{u}}^{n+1}, \rho_{\mathrm{fl}}\boldsymbol{u}_h^{n+1}\cdot\nabla\boldsymbol{v}_h + \mu_{\mathrm{fl}}\Delta\boldsymbol{v}_h + \nabla q_h \right\rangle_K - \sum_K \left\langle \tilde{p}^{n+1}, \nabla\cdot\boldsymbol{v}_h \right\rangle_K = L(\boldsymbol{V}_h), \quad (6)$$

where $\tilde{\boldsymbol{u}}^{n+1}$ and $\tilde{p}^{n+1}$ are the solution of

$$\frac{\rho_{\mathrm{fl}}}{\Delta t}\left(\tilde{\boldsymbol{u}}^{n+1} - \tilde{\boldsymbol{u}}^{n}\right) + \frac{1}{\tau_1}\tilde{\boldsymbol{u}}^{n+1} = \Pi^{\perp}\left(\boldsymbol{r}\left(\boldsymbol{U}_h^{n+1}\right)\right), \quad (7)$$

$$\frac{1}{\tau_2}\tilde{p}^{n+1} = -\Pi^{\perp}\left(\nabla\cdot\boldsymbol{u}_h^{n+1}\right), \quad (8)$$

within each element domain K of the finite element partition of the fluid domain, with

$$\boldsymbol{r}\left(\boldsymbol{U}_h^{n+1}\right) = \rho_{\mathrm{fl}}\delta_{2,t}\boldsymbol{u}_h^{n+1} - \mu_{\mathrm{fl}}\Delta\boldsymbol{u}_h^{n+1} + \rho_{\mathrm{fl}}\boldsymbol{u}_h^{n+1}\cdot\nabla\boldsymbol{u}_h^{n+1} + \nabla p_h^{n+1} - \rho_{\mathrm{fl}}\boldsymbol{f}^{n+1},$$

$$\tau_1 = \left[c_1\frac{\mu_{\mathrm{fl}}}{h^2} + c_2\frac{\rho_{\mathrm{fl}}|\boldsymbol{u}_h|_K}{h}\right]^{-1}, \quad (9)$$

$$\tau_2 = c_3\frac{h^2}{\tau_1}. \quad (10)$$

In Eqs. (7)-(8), $\Pi^{\perp}$ is the projection orthogonal to the finite element space (either of velocities or of pressures), computed as $\Pi^{\perp} = I - \Pi$, Π being the projection onto the adequate finite element space. In Eqs. (9)-(10), $|\boldsymbol{u}_h|_K$ is the mean velocity modulus in element K, h is the element size and c_1, c_2 and c_3 are stabilization constants. For linear elements we take $c_1 = 4.0$, $c_2 = 2.0$ and $c_3 = 1.0$; for quadratic elements we use the same values but taking h half the element size (roughly the distance between nodes of the element), as justified in [15].

The method we use was introduced in [16]. It belongs to the VMS family, with the sub-grid scales, i.e., the components of the unknowns that cannot be reproduced by the finite element space, orthogonal to it and time dependent; in fact, Eq. (7) is an approximation to the exact sub-grid scale equation using the backward Euler scheme to discretize in time.

3. NON-LINEAR SOLID ELASTO-DYNAMICS

In this section, a short review of non-linear solid elasto-dynamics formulation we employ is given, as well as the spatial and temporal discretization schemes used.

Prepared using nmeauth.cls
DOI: 10.1002/nme

3.1. Governing equations

Let Ω_s be the domain of the solid, with boundary $\Gamma_\mathrm{s} = \Gamma_{\mathrm{D,s}} \cup \Gamma_{\mathrm{N,s}}$, where $\Gamma_{\mathrm{D,s}}$ and $\Gamma_{\mathrm{N,s}}$ are boundaries where Dirichlet and Neumann conditions are prescribed, respectively. The time interval of analysis $]0, t_\mathrm{f}[$ is the same as for the fluid. The elasto-dynamics problem written in updated Lagrangian form (see for example [11]) consists in finding a displacement field $\boldsymbol{d} : \Omega_\mathrm{s} \times]0, t_\mathrm{f}[\longrightarrow \mathbb{R}^d$ such that:

$$\begin{aligned} \rho_\mathrm{s} \partial_{tt} \boldsymbol{d} - \nabla \cdot \boldsymbol{\sigma}_\mathrm{s} &= \rho_\mathrm{s} \boldsymbol{f} && \text{in } \Omega_\mathrm{s}, && t \in]0, t_\mathrm{f}[, \\ \boldsymbol{d} &= \boldsymbol{d}_\mathrm{D} && \text{on } \Gamma_{\mathrm{D,s}}, && t \in]0, t_\mathrm{f}[, \\ \boldsymbol{n}_\mathrm{s} \cdot \boldsymbol{\sigma}_\mathrm{s} &= \boldsymbol{t}_\mathrm{s} && \text{on } \Gamma_{\mathrm{N,s}}, && t \in]0, t_\mathrm{f}[, \\ \partial_t \boldsymbol{d} &= \dot{\boldsymbol{d}}^0 && \text{in } \Omega_\mathrm{s}, && t = 0, \\ \boldsymbol{d} &= \boldsymbol{d}^0 && \text{in } \Omega_\mathrm{s}, && t = 0, \end{aligned} \tag{11}$$

where ρ_s is the solid's density, $\boldsymbol{\sigma}_\mathrm{s}$ is the solid's Cauchy stress tensor, $\boldsymbol{f}$ is the acceleration vector of the solid (now defined on Ω_s), $\boldsymbol{d}^0$ is a prescribed initial displacement and $\dot{\boldsymbol{d}}^0$ is a prescribed initial velocity, $\boldsymbol{d}_D$ is a prescribed displacement on the boundary $\Gamma_{\mathrm{D,s}}$, $\boldsymbol{t}_\mathrm{s}$ is a prescribed traction on the boundary $\Gamma_{\mathrm{N,s}}$, and $\boldsymbol{n}_\mathrm{s}$ is the normal to the solid domain.

In the non-linear setting, the constitutive equation for the stress tensor can be modeled in a variety of ways and depends on the material to be simulated. In the present case we are interested in the Neo-Hookean and Saint Venant-Kirchoff material models, which can be defined as follows:

$$\begin{aligned} \text{NeoHookean: } \boldsymbol{\sigma}_\mathrm{s} &= \frac{1}{J} \left[\lambda_\mathrm{s} \ln(J) \boldsymbol{I} + \mu_\mathrm{s} (\boldsymbol{b} - \boldsymbol{I}) \right], \\ \text{Saint Venant-Kirchoff: } \boldsymbol{\sigma}_\mathrm{s} &= \frac{1}{J} \boldsymbol{F} \left[\lambda_\mathrm{s} \mathrm{tr}(\boldsymbol{E}) \boldsymbol{I} + 2\mu_\mathrm{s} \boldsymbol{E} \right] \boldsymbol{F}^T, \end{aligned}$$

where $\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}}$ is the deformation gradient, $J = \det(\boldsymbol{F})$, λ_s and μ_s are Lamé's parameters, $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T$ is the left Cauchy tensor, $\boldsymbol{I}$ is the identity tensor and $\boldsymbol{E}$ is the Cauchy-Green strain tensor.

3.2. Weak form

Let $\mathcal{E}_0 = \{\boldsymbol{e} \in H^1(\Omega_\mathrm{s})^d \mid \boldsymbol{e}|_{\Gamma_\mathrm{D}} = \boldsymbol{0}\}$ and $\mathcal{E}_\mathrm{D} \subset \{\boldsymbol{e} \in H^1(\Omega_\mathrm{s})^d \mid \boldsymbol{e}|_{\Gamma_\mathrm{D}} = \boldsymbol{d}_\mathrm{D}\}$ be the appropriate spaces where the test functions and the displacement field (for $t \in]0, t_\mathrm{f}[$) should belong, respectively. The weak form of the solid elasto-dynamics problem consists in finding $\boldsymbol{d}$ in an adequate subspace of $L^2(0, t_\mathrm{f}; \mathcal{E}_\mathrm{D})$ such that:

$$\begin{aligned} (\rho_\mathrm{s} \partial_{tt} \boldsymbol{d}, \boldsymbol{e}) - (\boldsymbol{\sigma}_\mathrm{s}, \nabla^s \boldsymbol{e}) &= \langle \rho_\mathrm{s} \boldsymbol{f}, \boldsymbol{e} \rangle + \langle \boldsymbol{t}_\mathrm{s}, \boldsymbol{e} \rangle_{\Gamma_{\mathrm{N,s}}}, && t \in]0, t_\mathrm{f}[, \\ (\partial_t \boldsymbol{d}, \boldsymbol{e}) &= (\dot{\boldsymbol{d}}^0, \boldsymbol{e}), && t = 0, \\ (\boldsymbol{d}, \boldsymbol{e}) &= (\boldsymbol{d}^0, \boldsymbol{e}), && t = 0, \end{aligned} \tag{12}$$

for all $\boldsymbol{e} \in \mathcal{E}_0$.

3.3. Spatial discretization

We can discretize the solid domain as done for the fluid, and use also an analogous notation. In this way we can now construct conforming finite element spaces $\mathcal{E}_{h,\mathrm{D}} \subset \mathcal{E}_\mathrm{D}$ and $\mathcal{E}_{h,0} \subset \mathcal{E}_0$. Then, the Galerkin finite element approximation can be written as finding $\boldsymbol{d}_h$ in the appropriate subspace of $L^2(0, t_\mathrm{f}; \mathcal{E}_{h,\mathrm{D}})$ such that:

$$\begin{aligned} (\rho_\mathrm{s} \partial_{tt} \boldsymbol{d}_h, \boldsymbol{e}_h) - (\boldsymbol{\sigma}_{\mathrm{s},h}, \nabla^s \boldsymbol{e}_h) &= \langle \rho_\mathrm{s} \boldsymbol{f}, \boldsymbol{e}_h \rangle + \langle \boldsymbol{t}_\mathrm{s}, \boldsymbol{e}_h \rangle_{\Gamma_{\mathrm{N,s}}}, && t \in]0, t_\mathrm{f}[, \\ (\partial_t \boldsymbol{d}_h, \boldsymbol{e}_h) &= (\dot{\boldsymbol{d}}^0, \boldsymbol{e}_h), && t = 0, \end{aligned} \tag{13}$$

Prepared using nmeauth.cls
DOI: 10.1002/nme

$$(\boldsymbol{d}_h, \boldsymbol{e}_h) = (\boldsymbol{d}^0, \boldsymbol{e}_h), \qquad\qquad t = 0,$$

for all $\boldsymbol{e}_h \in \mathcal{E}_{h,0}$, where $\boldsymbol{\sigma}_{\mathrm{s},h}$ is the Cauchy stress tensor evaluated with $\boldsymbol{d}_h$. This problem can be linearized using a Newton-Raphson scheme; for further details see for example [11].

3.4. Time discretization

Even though any finite difference scheme in time could be used, including the popular Newmark scheme, in this work for the temporal discretization the following second order Backward Differences scheme (BDF2) has been used:

$$\boldsymbol{a}^{n+1} = \frac{1}{(\Delta t)^2}\left(2\boldsymbol{d}^{n+1} - 5\boldsymbol{d}^n + 4\boldsymbol{d}^{n-1} - \boldsymbol{d}^{n-2}\right),$$

where $\boldsymbol{d}^{n+1}$ and $\boldsymbol{a}^{n+1}$ are approximations to the position and acceleration vectors at time step $n+1$, respectively.

4. FLUID STRUCTURE INTERACTION

Once all ingredients have been identified, it is possible to detail the process of dealing with FSI problems. In this section we first express the FSI equations in weak form, and then we detail the FSI algorithm as well as the boundary relaxation scheme used.

4.1. Governing equations and weak form

The approach followed in this work can be taken as the traditional one in a broad sense, where an updated Lagrangian formulation is used to deal with the solid mechanics problem while the fluid problem is solved by means of an Arbitrary-Lagrangian-Eulerian (ALE) formulation (see for example [19]) to cope with the domain motion.

Borrowing the notation developed in previous sections, we can expand it to account for a moving domain and to take into account the interaction between sub-domains. We will use a superscript t to indicate the time dependency of the spaces and a temporal argument for the geometrical domains. For the FSI problem the space for the continuous problem can be defined as $\mathcal{F}_{D,t} = \mathcal{W}_{D,t} \times \mathcal{E}_{D,t}$, and the initial and boundary value problem can be stated as finding $[\boldsymbol{u}, p, \boldsymbol{d}] \in L^2(0, t_\mathrm{f}; \mathcal{V}_{D,t}) \times L^1(0, t_\mathrm{f}; \mathcal{Q}_{D,t}) \times L^2(0, t_\mathrm{f}; \mathcal{E}_{D,t})$ such that:

$$\begin{aligned}
&\rho_\mathrm{fl}(\partial_t \boldsymbol{u}, \boldsymbol{v}) - 2\mu_\mathrm{fl}(\nabla^s \boldsymbol{u}, \nabla^s \boldsymbol{v}) \\
&\quad +\rho_\mathrm{fl}\langle \boldsymbol{c}\cdot\nabla\boldsymbol{u}, \boldsymbol{v}\rangle - (p, \nabla\cdot\boldsymbol{v}) = \langle \rho_\mathrm{fl}\boldsymbol{f}, \boldsymbol{v}\rangle + \langle \boldsymbol{t}_\mathrm{fl}, \boldsymbol{v}\rangle_{\Gamma_{\mathrm{N,fl}}} \quad \text{in } \Omega(t)_\mathrm{fl},\ t \in]0, t_\mathrm{f}[, \\
&\qquad\qquad (q, \nabla\cdot\boldsymbol{u}) = 0 \quad \text{in } \Omega(t)_\mathrm{fl},\ t \in]0, t_\mathrm{f}[, \\
&\quad (\rho_\mathrm{s}\partial_{tt}\boldsymbol{d}, \boldsymbol{e}) - (\boldsymbol{\sigma}_\mathrm{s}, \nabla^s \boldsymbol{e}) = \langle \rho_\mathrm{s}\boldsymbol{f}, \boldsymbol{e}\rangle + \langle \boldsymbol{t}_\mathrm{s}, \boldsymbol{e}\rangle_{\Gamma_{\mathrm{N,s}}} \quad \text{in } \Omega(t)_\mathrm{s},\ t \in]0, t_\mathrm{f}[, \\
&\qquad\qquad \boldsymbol{u} = \partial_t \boldsymbol{d} \quad \text{on } \Gamma(t)_\mathrm{I},\ t \in]0, t_\mathrm{f}[, \\
&\qquad\quad \boldsymbol{n}_\mathrm{s}\cdot\boldsymbol{\sigma}_\mathrm{s} + \boldsymbol{n}_\mathrm{fl}\cdot\boldsymbol{\sigma}_\mathrm{fl} = \boldsymbol{0} \quad \text{on } \Gamma(t)_\mathrm{I},\ t \in]0, t_\mathrm{f}[,
\end{aligned} \tag{14}$$

for all $[\boldsymbol{v}, q, \boldsymbol{e}] \in \mathcal{F}_{0,t}$, with $\mathcal{F}_{0,t} = \mathcal{W}_{0,t} \times \mathcal{E}_{0,t}$, and satisfying initial conditions in a weak sense. In the momentum equation for the fluid, $\boldsymbol{c}$ is known as the convection velocity from the domain point of view, which is given by $\boldsymbol{c} = \boldsymbol{u} - \boldsymbol{u}_\mathrm{domain}$, where $\boldsymbol{u}_\mathrm{domain}$ is the velocity of the points in the computational domain to which the unknowns are referred. Note that in this form, the domains to which the fluid and solid pertain, $\Omega(t)_\mathrm{fl}$ and $\Omega(t)_\mathrm{s}$, respectively, are now time dependent, as they change according to the deformation process. The boundary $\Gamma(t)_\mathrm{I}$ is the interface for both domains, as shown in Figure 1.

Prepared using nmeauth.cls
DOI: 10.1002/nme

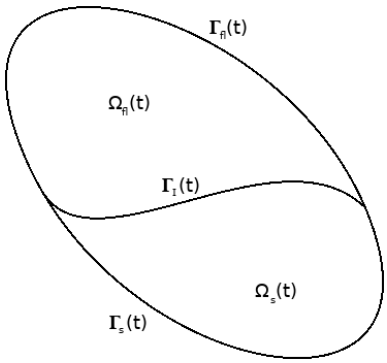


Figure 1. Domain composed of two different sub-domains, $\Omega_{\mathrm{fl}}(t)$ and $\Omega_{\mathrm{s}}(t)$, their interface $\Gamma_{\mathrm{I}}(t)$ and respective boundaries $\Gamma_{\mathrm{fl}}(t)$ and $\Gamma_{\mathrm{s}}(t)$

The finite element approximation and integration in time of problem (14) follows the lines described in Sections 2 and 3 for each subproblem, and we shall not detail this further. It only remains to indicate how to treat numerically the transmission conditions, and this depends on the coupling scheme described next.

From now on, we will consider that the unknowns have been discretized, although we shall not introduce the subscript h to lighten the notation. In particular, we shall refer to $\boldsymbol{u}_{\mathrm{domain}}$ as the mesh velocity, $\boldsymbol{u}_{\mathrm{mesh}}$, as the points in the domain will be moved according to the velocity computed for the mesh nodes. The mesh movement algorithm that we have employed has been taken from [13], which has proven simple, robust and reliable. The only restriction that $\boldsymbol{u}_{\mathrm{mesh}}$ has to satisfy is that it must be equal to the velocity of the boundary of the fluid domain at this boundary.

4.2. Coupling scheme

The are various ways to treat the numerical system for the interaction problem regardless of the particular formulation used to solve each domain. In a monolithic coupling the whole problem is assembled and solved, coupling is treated implicitly (see for example [21, 34]). This approach benefits from increased stability on the solution but requires a solver specially tailored for coupled FSI problems. On the other hand, partitioned approaches assemble each domain independently and coupling is achieved through right-hand-side terms of each system that need to be guessed. For strongly coupled systems, sub-iterations, and very often relaxation, are necessary to guarantee convergence on the interaction boundaries. In some cases, a high number of coupling iterations are necessary to achieve convergence (see [1, 31]). Finally, a less popular approach is to use a staggered coupling (or loosely coupled interaction); this is essentially a partitioned approach where the boundary conditions are treated explicitly and no sub-iterations are done. This approach can suffer from instabilities, like the added mass effect (see for example [24]).

In this work we employ a partitioned strongly coupled scheme to achieve domain coupling; this means that for every time-step each domain is iterated independently until convergence is achieved for velocity, pressure and displacement on the interaction boundary. This creates the necessity of an additional convergence block that guarantees coupling convergence. In total we are left with three coupling blocks, these being the internal solver convergence, the non-linearity convergence of each problem (fluid and solid) and the coupling convergence for the interaction boundary. This is clarified in Section 4.3. In our implementation, iteration by sub-domain can be done for non matching meshes by means of the Lagrange interpolation functions to ensure continuity of certain quantities, as shown in [28].

The order in which we iterate is the standard one, the one that guarantees stability of the process due to the different ‘stiffnesses’ of the sub-problems, namely, a Dirichlet-Neumann coupling. The way

Prepared using nmeauth.cls
DOI: 10.1002/nme

to proceed is to determine the shape of the fluid domain from the deformation obtained at a certain iteration within a given time step for the solid, as well as the velocity of the domain boundary; from this, one can compute the velocity of the mesh in the fluid domain and solve the flow equations. Once velocities and pressures in the fluid are computed, the resulting normal stress on the solid boundary can be obtained, and this can be used to solve the problem in the solid domain. The process needs to be repeated until convergence is achieved. This guarantees that both transmission conditions in problem (14) will hold.

In the process described, relaxation of the transmitted quantities is very often required if not mandatory. This allows one to minimize the number of block (fluid and solid) iterations. In this respect, we have used a relaxation of the position and velocity of the interface boundary that the solid solver transmits to the fluid solver. We denote this position as $\boldsymbol{d}_{\Gamma_\mathrm{I}}$; from it, one may compute the velocity of the fluid boundary and $\boldsymbol{u}_{\mathrm{mesh}}$, as explained above. We have implemented an Aitken relaxation scheme, in particular Aitken Δ^2, detailed in [30], which we describe now in our context. Within each time step, let us denote by a superscript k the k-th block-iteration of any variable. For clarity, let us omit the superscript with the time step counter. Suppose that from values at the k-th iteration, the solid is solved, obtaining the boundary displacements $\boldsymbol{d}^{k+1}_{\Gamma_\mathrm{I},\mathrm{s}}$. Then, the fluid is solved from the boundary displacements $\boldsymbol{d}^{k+1}_{\Gamma_\mathrm{I}}$ computed as

$$\boldsymbol{d}^{k+1}_{\Gamma_\mathrm{I}} = \boldsymbol{d}^{k}_{\Gamma_\mathrm{I}} + \omega^{k+1}\boldsymbol{r}^{k+1}_{\Gamma_\mathrm{I}}, \tag{15}$$

where

$$\boldsymbol{r}^{k+1}_{\Gamma_\mathrm{I}} := \boldsymbol{d}^{k+1}_{\Gamma_\mathrm{I},\mathrm{s}} - \boldsymbol{d}^{k}_{\Gamma_\mathrm{I}}, \tag{16}$$

$$\omega^{k+1} = -\omega^{k}\frac{(\boldsymbol{r}^{k}_{\Gamma_\mathrm{I}})^T(\boldsymbol{r}^{k+1}_{\Gamma_\mathrm{I}} - \boldsymbol{r}^{k}_{\Gamma_\mathrm{I}})}{|\boldsymbol{r}^{k+1}_{\Gamma_\mathrm{I}} - \boldsymbol{r}^{k}_{\Gamma_\mathrm{I}}|^2}. \tag{17}$$

4.3. General FSI algorithm

For a time interval between 0 and t_f, let n be the current time step, n_{last} the last time step, i the current internal iteration of a particular sub-domain (fluid or solid), k the current coupling iteration for both domains, $\mathrm{Tol}_{\mathrm{time}}$ the temporal tolerance (to decide whether the steady state has been reached or nor), $\mathrm{Tol}_{\mathrm{cou}}$ the coupling tolerance between sub-domains, $\mathrm{Tol}_{\mathrm{s}}$ the internal tolerance for convergence for the solid sub-domain, and $\mathrm{Tol}_{\mathrm{fl}}$ the internal tolerance for convergence for the fluid sub-domain. The FSI algorithm is displayed in Algorithm 1. For $i = 0$ (either for the fluid or for the solid) the unknowns are initialized to those of the previous block-iteration, whereas for $k = 0$ they are initialized to those of the previous time step.

5. REDUCED ORDER MODELING

As discussed in Section 4.2, strongly coupled partitioned FSI algorithms may require a high number of sub-iterations and sub-relaxation, making the problem potentially expensive numerically and consequently taking a long time to achieve a solution. In this sense, the development of model order reduction schemes that increase performance while maintaining output accuracy is of interest. Herein lies our motivation to introduce ROM into FSI. In this section we give a short review of the methodology we apply and the algorithmic aspects that concern it.

5.1. Some ROM theory and notation

Let us define a high dimensional space $\mathcal{Y}_h$ of dimension M, with $\boldsymbol{\varphi} = \{\boldsymbol{\varphi}^1, \ldots, \boldsymbol{\varphi}^M\}$ its orthonormal basis, whose elements are vectors of D components. Then any element $\boldsymbol{y}_h \in \mathcal{Y}_h$ can be

DOI: 10.1002/nme

Algorithm 1 General FSI algorithm

Read case parameters and initialize values for the fluid and the solid domains
for $n = 1; n \leq n_{\text{last}}; n+1$ **do**
  for $k = 1; k \leq k_{\max}; k+1$ **do**
    for $i = 1; i \leq i_{\max}; i+1$ **do**
      Solve the fluid problem for $[\boldsymbol{u}_h, p_h]^{i+1}$ from $\boldsymbol{u}^k_{\text{mesh}}$
      Calculate error $\epsilon^{i+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}_h^{i+1} - \boldsymbol{u}_h^i|}{|\boldsymbol{u}_h^{i+1}|}$; $\epsilon^{i+1}_p = \frac{|p_h^{i+1} - p_h^i|}{|p_h^{i+1}|}$;
      if $\epsilon^{i+1}_{\boldsymbol{u}}$ and $\epsilon^{i+1}_{\boldsymbol{p}} \leq \text{Tol}_{\text{fl}}$ **then**
        Non linearity converged; break non-linearity loop
      end if
    end for
    Set $[\boldsymbol{u}_h, p_h]^{k+1} \leftarrow [\boldsymbol{u}_h, p_h]^{i+1}$
    Calculate tractions $\boldsymbol{t}^{k+1}_{\text{fl}}$ on Γ_{I} (to be transmitted to the solid)
    for $i = 1; i \leq i_{\max}; i+1$ **do**
      Solve the solid problem for $\boldsymbol{d}^{i+1}_h$ from $\boldsymbol{t}^{k+1}_{\text{fl}}$
      Calculate error $\epsilon^{i+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}_h^{i+1} - \boldsymbol{d}_h^i|}{|\boldsymbol{d}_h^{i+1}|}$
      if $\epsilon^{i+1}_{\boldsymbol{d}} \geq \text{Tol}_{\text{s}}$ **then**
        Non linearity converged; break non-linearity loop
      end if
    end for
    Set $\boldsymbol{d}^{k+1}_h \leftarrow \boldsymbol{d}^{i+1}_h$ and $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}},\text{s}}$ the values of $\boldsymbol{d}^{k+1}_h$ on Γ_{I}
    Calculate the residual on the $\boldsymbol{r}^{k+1}_{\Gamma_{\text{I}}}$ from (16)
    Calculate the relaxation parameter ω^{k+1} from (17)
    Calculate the mesh interface movement $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}}} = \boldsymbol{d}^k_{\Gamma_{\text{I}}} + \omega^{k+1}\boldsymbol{r}^{k+1}_{\Gamma_{\text{I}}}$
    Calculate the fluid mesh movement $\boldsymbol{d}^{k+1}_{\text{mesh}}$ and the fluid mesh velocity $\boldsymbol{u}^{k+1}_{\text{mesh}}$ from $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}}}$
    Calculate the coupling error $\epsilon^{k+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}_h^{k+1} - \boldsymbol{u}_h^k|}{|\boldsymbol{u}_h^{k+1}|}$; $\epsilon^{k+1}_p = \frac{|p_h^{k+1} - p_h^k|}{|p_h^{k+1}|}$; $\epsilon^{k+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}_h^{k+1} - \boldsymbol{d}_h^k|}{|\boldsymbol{d}_h^{k+1}|}$
    if $\epsilon^{k+1}_{\boldsymbol{u}}$ and ϵ^{k+1}_p and $\epsilon^{k+1}_{\boldsymbol{d}}$ on $\Gamma_{\text{I}} \leq \text{Tol}_{\text{coup}}$ **then**
      Coupling converged; break coupling loop
    end if
  end for
  Calculate temporal increment $\epsilon^{n+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^n|}{|\boldsymbol{u}_h^{n+1}|}$; $\epsilon^{n+1}_p = \frac{|p_h^{n+1} - p_h^n|}{|p_h^{n+1}|}$; $\epsilon^{n+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}_h^{n+1} - \boldsymbol{d}_h^n|}{|\boldsymbol{d}_h^{n+1}|}$
  if $\epsilon^{n+1}_{\boldsymbol{u}}$ and ϵ^{n+1}_p and $\epsilon^{n+1}_{\boldsymbol{d}} \leq \text{Tol}_{\text{time}}$ **then**
    Stationary state achieved; break temporal loop
  end if
end for
Finalize case
Output if necessary

written as the linear combination $\boldsymbol{y}_h = \sum_{k=1}^{M} (\boldsymbol{y}_h, \boldsymbol{\varphi}^k)\boldsymbol{\varphi}^k$, now with $(\cdot,\cdot)$ the L^2-inner product in $\mathcal{Y}_h$. We can also define a low-dimensional subspace $\mathcal{Y}_{\text{rom}} \subset \mathcal{Y}_h$ of dimension m, which approximates $\mathcal{Y}_h$ as $m \to M$, with a basis $\boldsymbol{\phi} = \{\boldsymbol{\phi}^1, \ldots, \boldsymbol{\phi}^m\}$. Using this basis, we can approximate any element $\boldsymbol{y}_h$ as $\boldsymbol{y}_h \approx \boldsymbol{y}_{\text{rom}} = \sum_{k=1}^{m} \boldsymbol{\phi}^k a^k$, where a^k is the k-th coefficient which can be computed as $a^k = (\boldsymbol{y}_h, \boldsymbol{\phi}^k)$, and which will typically be obtained from the solution of the reduced problem. The accuracy of the approximation depends on how well the basis $\boldsymbol{\phi}$ approximates the exact basis $\boldsymbol{\varphi}$.

5.1.1. Construction of the basis The method we use to construct the basis of the low-dimensional space is the Proper Orthogonal Decomposition (POD). The objective of this method is finding a

Prepared using nmeauth.cls
DOI: 10.1002/nme

basis from a collection of high-fidelity "snapshots", which in our context are solutions in $\mathcal{Y}_h$ of an evolution problem at certain time steps. As it is well known, we need to substract the mean. Thus, taking a set of data as a collection of N snapshots $\{\boldsymbol{s}_j\}_{j=1}^N = \{\boldsymbol{y}_{h,j} - \bar{\boldsymbol{y}}_h\}_{j=1}^N$, the overbar denoting the mean of the $\boldsymbol{y}_{h,j} \in \mathcal{Y}_h$, we can reproduce any element of this collection as

$$\boldsymbol{y}_{h,j} \approx \bar{\boldsymbol{y}}_h + \sum_{k=1}^{m} (\boldsymbol{s}_j, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k, \tag{18}$$

where, in the case of POD, $\{\boldsymbol{\phi}^k\}_{k=1}^m$ is an orthonormal system of $\mathcal{Y}_h$. The POD consists in finding the orthonormal basis $\{\boldsymbol{\phi}^k\}_{k=1}^m$ of $\mathcal{Y}_{\text{rom}}$ such that:

$$\begin{aligned} \min_{\{\boldsymbol{\phi}\}_{k=1}^m} \quad & \frac{1}{N}\sum_{j=1}^{N} \left\| \boldsymbol{s}_j - \sum_{k=1}^{m} (\boldsymbol{s}_j, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k \right\|^2, \\ \text{subject to } & (\boldsymbol{\phi}^i, \boldsymbol{\phi}^j) = \delta_{ij}, \quad 1 \le i, j \le m. \end{aligned} \tag{19}$$

By means of a Singular Value Decomposition (SVD) we can solve for the basis $\{\boldsymbol{\phi}^k\}_{k=1}^m$ from the matrix of snapshots. This basis depends on parameters as time-step, how often the snapshots were acquired and the reproducibility of the function being analyzed. A reduced basis can be defined by truncating the left singular-vectors at the m-th column. As a criterion for the truncation, we use the retained energy η, defined in [36] as:

$$\eta = \frac{\sum_{k=1}^{m} \lambda^k}{\sum_{k=1}^{M} \lambda^k}, \tag{20}$$

where $\{\lambda^k\}_{k=1}^M$ are the singular values of the SVD. The SVD produces a diagonal matrix which contains, from greatest to smallest, the eigenvalues of the associated (singular) eigenvectors. The ordering of the eigenvalues is a measure of the relative importance of each of the basis functions in the whole system. In general, in a reducible problem (a problem that should be easily reproduced by means of ROM) they decrease quickly in magnitude. If m is sufficiently small, the time to compute the reduced system is minimal.

The stage of the problem in which the basis is calculated is termed as the off-line phase.

Remark 1
In the rest of this work, a basis which contains a greater number of vectors than another one will be referred as a 'richer' basis. In fact, it is the basis of a higher dimensional space.

Remark 2
The snapshots are arrays of M components, and therefore the vectors of the ROM basis are also arrays of M components. In our finite element context, however, we may identify them as piecewise polynomial functions. Indeed, if $\phi^{k,a}$ is the a-th component of the k-th basis vector, $a = 1, \ldots, M$, we may identify $\boldsymbol{\phi}^k$ with the function

$$\phi^k(\boldsymbol{x}) = \sum_{a=1}^{M} N^a(\boldsymbol{x})\phi^{k,a}, \ k = 1, \ldots, m,$$

where $\boldsymbol{x}$ is the position vector and $N^a(\boldsymbol{x})$ the finite element interpolation function of the a-th degree of freedom. Therefore, $\mathcal{Y}_{\text{rom}}$ can be identified as a space of functions of dimension m.

In the case of FSI problems, the construction of the basis can be done in a variety of ways, as shown in [4] for domain decomposition problems. In our case, and as a first approach, it was decided to assemble and calculate the snapshots of each sub-domain separately, this is, the basis

DOI: 10.1002/nme

for the fluid domain from the snapshots of velocity and pressure in the fluid ($\boldsymbol{\phi}_{\mathrm{fl}}(\boldsymbol{u}_h, p_h)$) and the basis for the solid domain from the the snapshots of displacements ($\boldsymbol{\phi}_{\mathrm{s}}(\boldsymbol{d}_h)$). However, it is also possible to construct and assemble just one basis from the joint snapshots of velocity, pressure and displacement ($\boldsymbol{\phi}_{\mathrm{fl,s}}(\boldsymbol{u}_h, p_h, \boldsymbol{d}_h)$); the performance of this option is left for future study (see [4] for further details).

5.1.2. VMS-ROM In FSI problems and with the partitioned strategy we have followed, we have to solve one variational problem in the fluid domain and another one in the solid domain. The Galerkin finite element approximation to these problems can be stated as these variational problems restricted to the finite element spaces, both for the unknowns and for the test functions. In a similar way, the ROM could be expressed as the same variational problems, now restricting unknowns and test functions to the ROM spaces, with a much smaller dimension than the finite element spaces. There are other options to state the ROM problem, but the approach described justifies that one may expect similar instability problems for the Galerkin finite element method and the described Galerkin-ROM. Therefore, some sort of numerical stabilization will be required *for the fluid problem*. The solid problem we have considered (without incompressibility of mixed formulations) does not require any stabilization, the Galerkin method yields a stable and accurate approximation.

The approach we shall follow for the fluid is the same as for the finite element problem, namely, to use a VMS method with an approximation to the sub-grid scales similar to that given by (7). In the ROM case, it is particularly natural to use *orthogonal* sub-grid scales, since the vectors of the basis are mutually orthogonal. Therefore, if the ROM space is obtained by truncating the vectors obtained from a SVD of the collection of snapshots to the first m members, the space of sub-grid scales is simply its L^2-orthogonal complement.

Let $\boldsymbol{U}_{\mathrm{rom}} \equiv [\boldsymbol{u}_{\mathrm{rom}}, p_{\mathrm{rom}}] :]0, t_{\mathrm{f}}[\longrightarrow \mathcal{Y}_{\mathrm{rom}}$ and $\boldsymbol{V}_{\mathrm{rom}} \equiv [\boldsymbol{v}_{\mathrm{rom}}, q_{\mathrm{rom}}] \in \mathcal{Y}_{\mathrm{rom}}$ be the ROM unknown and test functions of the fluid problem, respectively, where $\mathcal{Y}_{\mathrm{rom}}$ is the velocity-pressure pair obtained from the POD basis (using its interpretation as a function space described in Remark 2). According to the previous considerations, and using the BDF2 scheme for the time discretization, the problem we have to solve for the ROM is:

$$
\begin{aligned}
&(\rho_{\mathrm{fl}}\delta_{2,t}\boldsymbol{u}_{\mathrm{rom}}^{n+1}, \boldsymbol{v}_h) + B(\boldsymbol{U}_{\mathrm{rom}}^{n+1}, \boldsymbol{V}_h) + \sum_K \left\langle \breve{\boldsymbol{u}}^{n+1}, \rho_{\mathrm{fl}}\boldsymbol{u}_{\mathrm{rom}}^{n+1}\cdot\nabla\boldsymbol{v}_{\mathrm{rom}} + \mu_{\mathrm{fl}}\Delta\boldsymbol{v}_{\mathrm{rom}} + \nabla q_{\mathrm{rom}}\right\rangle_K \\
&\quad - \sum_K \left\langle \breve{p}^{n+1}, \nabla\cdot\boldsymbol{v}_{\mathrm{rom}}\right\rangle_K = L(\boldsymbol{V}_{\mathrm{rom}}),
\end{aligned}
\tag{21}
$$

where $\breve{\boldsymbol{u}}^{n+1}$ and $\breve{p}^{n+1}$ are the solution of

$$
\frac{\rho_{\mathrm{fl}}}{\Delta t}\left(\breve{\boldsymbol{u}}^{n+1} - \breve{\boldsymbol{u}}^{n}\right) + \frac{1}{\tau_1}\breve{\boldsymbol{u}}^{n+1} = \Pi_{\mathrm{rom}}^{\perp}\left(\boldsymbol{r}\left(\boldsymbol{U}_{\mathrm{rom}}^{n+1}\right)\right), \tag{22}
$$

$$
\frac{1}{\tau_2}\breve{p}^{n+1} = -\Pi_{\mathrm{rom}}^{\perp}\left(\nabla\cdot\boldsymbol{u}_{\mathrm{rom}}^{n+1}\right), \tag{23}
$$

within each element domain K of the finite element partition of the fluid domain, with

$$
\boldsymbol{r}\left(\boldsymbol{U}_{\mathrm{rom}}^{n+1}\right) = \rho_{\mathrm{fl}}\delta_{2,t}\boldsymbol{u}_{\mathrm{rom}}^{n+1} - \mu_{\mathrm{fl}}\Delta\boldsymbol{u}_{\mathrm{rom}}^{n+1} + \rho_{\mathrm{fl}}\boldsymbol{u}_{\mathrm{rom}}^{n+1}\cdot\nabla\boldsymbol{u}_{\mathrm{rom}}^{n+1} + \nabla p_{\mathrm{rom}}^{n+1} - \rho_{\mathrm{fl}}\boldsymbol{f}^{n+1},
$$

and the stabilization parameters computed as in (9)-(10), replacing $\boldsymbol{u}_h$ by $\boldsymbol{u}_{\mathrm{rom}}$ in the former. In (22)-(23), $\Pi_{\mathrm{rom}}^{\perp} = I - \Pi_{\mathrm{rom}}$, where Π_{rom} is the L^2-projection onto the appropriate ROM space (of velocities or of pressures).

Remark 3
Note that we make use of the finite element partition, both in problem (21) and in the definition of the stabilization parameters. This is possible because the ROM basis vectors can be understood as piecewise polynomial functions defined on each element of the partition (see Remark 2).

DOI: 10.1002/nme

Remark 4
Contrary to [6], the space where the subscales belong is directly the L^2-orthogonal to the ROM space, whereas in the cited reference it is a subspace of the finite element space.

As a concluding remark of this section we would like to address our choice of not using modal analysis based methods, which is usually the norm in ROM for solids. It is clear that fluid flow is impossible to be represented via this kind of eigenvalue decomposition, specially for the highly non-linear nature of the flows we are interested in. Even though it is possible to represent the non-linearities present in structural dynamics by modal analysis, as it is mentioned in [37], a basis calculated by this approach needs to be recalculated every so often to guarantee that the solution will reproduce accurately non-linear behavior. Our approach focuses on the idea of "one-for-all" methodology, where by means of one robust formulation any kind of problem can be represented. In conclusion, we apply the same form of decomposition (namely, POD) to both the fluid and the structure.

5.2. The algorithm

We describe next the algorithm to solve FSI problems using ROM for both the fluid and the solid, which we denote as ROM-ROM algorithm. However, we first describe the modifications that need to be done in the full order model, that we denote FOM-FOM algorithm, to obtain the necessary data for the ROM-ROM case.

5.2.1. FOM-FOM case Algorithm 2, essentially the same as in Algorithm 1 but with minor differences, corresponds to the off-line phase of a simulation case. We make use of all variables and parameters previously defined and add $\boldsymbol{\phi}_{\text{fl}}$ and $\boldsymbol{\phi}_{\text{s}}$; these are the fluid and solid basis, respectively. The dots represent the parts that are the same as in Algorithm 1.

Algorithm 2 FOM-FOM algorithm previous to a ROM-ROM calculation

```
Read case parameters and initialize values for fluid and solid domains, number of snapshots to
take and parameters for the SVD solver
for n = 1; n ≤ n_last; n + 1 do
  for k = 1; k ≤ k_max; k + 1 do
    ...
    for i = 1; i ≤ i_max; i + 1 do
      Solve fluid domain ...
    end for
    for i = 1; i ≤ i_max; i + 1 do
      Solve solid domain ...
    end for
    ...
    if ε_u^{k+1} and ε_p^{k+1} and ε_d^{k+1} on Γ_I ≤ Tol_coup then
      Store snapshot of u^{n+1}, p^{n+1} if required
      Store snapshot of d^{n+1} if required
      Coupling converged; Break Coupling loop
    end if
  end for
  ...
end for
Finalize case
Calculate bases φ_fl, φ_s by solving problem (19)
Output if necessary
```


DOI: 10.1002/nme

Remark 5
This process is most efficiently done taking full advantage of parallel solving, both for the FOM and ROM versions of the case. This means that the basis can be calculated and written to disk in parallel as well.

5.2.2. ROM-ROM case Algorithm 3 shows the ROM phase for the coupled problem, also known as the on-line phase. We make use of all the parameters defined in Section 5.2.1 and add $[\bar{\boldsymbol{u}}, \bar{p}, \bar{\boldsymbol{d}}]$, which are the snapshot mean values for the fluid and solid unkwnons (velocity-pressure and displacements). In essence, it is the same as Algorithm 1 replacing finite element unknowns by ROM unknowns.

6. NUMERICAL RESULTS

In this section, results of the FSI-ROM-ROM formulation proposed are shown. Two problems have been analyzed which exemplify two cases of interest, this being a semi-stationary case and a fully transient FSI case. For the fluid domain generally plots of integral quantities are preferred (lift and/or drag), pressure in the case of the semi-stationary case. For the solid domain displacement and acceleration plots are usually shown. A Fourier transform of the results is presented whenever deemed necessary.

Regarding the ROM problem, results are presented comparing the ROM result with the FOM result for the same case, using the basis that produced the most accurate results. Basis energy percentage is used for comparison between ROM results. Notice that in this work, each case consists of two reduced problems, for the solid and for the fluid, each with its particular basis taken from a different amount of snapshots.

It was observed that the ROM cases require a greater stabilization of the incompressibility term than the FOM cases to guarantee accurate results. For all FOM cases the algorithmic constant in Eq. (10) has been taken as $c_3 = 1.0$. In Section 6.2 we explore the effect of a slight variation of this constant.

6.1. Semi-stationary bending of FSI plate

This 2D semi-stationary problem, taken from [3], consists of a clamped plate perpendicular to the fluid flow. Once the flow starts from the left wall it will bend the plate. For the particular conditions of the test, a steady state for the plate is achieved, in which it is bent without oscillating. In the fluid, only the vortices created at the tip of the plate are transient.

The test conditions are shown in Table I (SI units are assumed everywhere). In this example, all ROM cases have been solved using the constant $c_3 = 2.0$ in the ROM equation corresponding to Eq. (10).

Table I. Physical parameters

	Fluid		Solid
ρ_{fl}	2.0	ρ_{s}	10.0
μ_{fl}	0.2	μ_{s}	5,000
		λ_{s}	2,000
Model	Newtonian		Neo-Hookean

Figure 2 shows the geometry and mesh for the test, where $H = 20$, $L = 80$, $h = 1$, $l = 10$.

Table II shows important mesh parameters and Table III the boundary conditions.

DOI: 10.1002/nme

Algorithm 3 ROM-ROM algorithm

Read case parameters and initialize values for fluid and solid domains
Initialize Fluid problem: read previously calculated reduced basis $\boldsymbol{\phi}_{\text{fl}}$, and select the desired amount of basis vectors through any criteriom (energy for example).
Initialize Solid problem: read previously calculated reduced basis $\boldsymbol{\phi}_{\text{s}}$, and select the desired amount of basis vectors through any criteriom (energy for example).
for $n = 1; n \leq n_{\text{last}}; n+1$ **do**
  for $k = 1; k \leq k_{\text{max}}; k+1$ **do**
    for $i = 1; i \leq i_{\text{max}}; i+1$ **do**
      Write $[\boldsymbol{u}_{\text{rom}}, p_{\text{rom}}]^{i+1}$ in terms of $\boldsymbol{\phi}_{\text{fl}}$ and $[\bar{\boldsymbol{u}}, \bar{p}]$
      Solve the fluid problem for $[\boldsymbol{u}_{\text{rom}}, p_{\text{rom}}]^{i+1}$ from $\boldsymbol{u}^k_{\text{mesh}}$
      Calculate error $\epsilon^{i+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}^{i+1}_{\text{rom}} - \boldsymbol{u}^{i}_{\text{rom}}|}{|\boldsymbol{u}^{i+1}_{\text{rom}}|}$; $\epsilon^{i+1}_{p} = \frac{|p^{i+1}_{\text{rom}} - p^{i}_{\text{rom}}|}{|p^{i+1}_{\text{rom}}|}$;
      if $\epsilon^{i+1}_{\boldsymbol{u}}$ and $\epsilon^{i+1}_{\boldsymbol{p}} \leq \text{Tol}_{\text{fl}}$ **then**
        Non linearity converged; break non-linearity loop
      end if
    end for
    Set $[\boldsymbol{u}_{\text{rom}}, p_{\text{rom}}]^{k+1} \leftarrow [\boldsymbol{u}_{\text{rom}}, p_{\text{rom}}]^{i+1}$
    Calculate tractions $\boldsymbol{t}^{k+1}_{\text{fl}}$ on Γ_{I} (to be transmitted to the solid)
    for $i = 1; i \leq i_{\text{max}}; i+1$ **do**
      Write $\boldsymbol{d}^{i+1}_{\text{rom}}$ in terms of $\boldsymbol{\phi}_{\text{s}}$ and $\bar{\boldsymbol{d}}$
      Solve the solid problem for $\boldsymbol{d}^{i+1}_{\text{rom}}$ from $\boldsymbol{t}^{k+1}_{\text{fl}}$
      Calculate error $\epsilon^{i+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}^{i+1}_{\text{rom}} - \boldsymbol{d}^{i}_{\text{rom}}|}{|\boldsymbol{d}^{i+1}_{\text{rom}}|}$
      if $\epsilon^{i+1}_{\boldsymbol{d}} \geq \text{Tol}_{\text{s}}$ **then**
        Non linearity converged; break non-linearity loop
      end if
    end for
    Set $\boldsymbol{d}^{k+1}_{\text{rom}} \leftarrow \boldsymbol{d}^{i+1}_{\text{rom}}$ and $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}},\text{s}}$ the values of $\boldsymbol{d}^{k+1}_{\text{rom}}$ on Γ_{I}
    Calculate the residual on the $\boldsymbol{r}^{k+1}_{\Gamma_{\text{I}}}$ from (16)
    Calculate relaxation parameter ω^{k+1} from (17)
    Calculate mesh interface movement $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}}} = \boldsymbol{d}^{k}_{\Gamma_{\text{I}}} + \omega^{k+1}\boldsymbol{r}^{k+1}_{\Gamma_{\text{I}}}$
    Calculate the fluid mesh movement $\boldsymbol{d}^{k+1}_{\text{mesh}}$ and fluid mesh velocity $\boldsymbol{u}^{k+1}_{\text{mesh}}$ from $\boldsymbol{d}^{k+1}_{\Gamma_{\text{I}}}$
    Calculate coupling error $\epsilon^{k+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}^{k+1}_{\text{rom}} - \boldsymbol{u}^{k}_{\text{rom}}|}{|\boldsymbol{u}^{k+1}_{\text{rom}}|}$; $\epsilon^{k+1}_{p} = \frac{|p^{k+1}_{\text{rom}} - p^{k}_{\text{rom}}|}{|p^{k+1}_{\text{rom}}|}$; $\epsilon^{k+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}^{k+1}_{\text{rom}} - \boldsymbol{d}^{k}_{\text{rom}}|}{|\boldsymbol{d}^{k+1}_{\text{rom}}|}$
    if $\epsilon^{k+1}_{\boldsymbol{u}}$ and ϵ^{k+1}_{p} and $\epsilon^{k+1}_{\boldsymbol{d}}$ on $\Gamma_{\text{I}} \leq \text{Tol}_{\text{coup}}$ **then**
      Coupling converged; break coupling loop
    end if
  end for
  Calculate temporal increment $\epsilon^{n+1}_{\boldsymbol{u}} = \frac{|\boldsymbol{u}^{n+1}_{\text{rom}} - \boldsymbol{u}^{n}_{\text{rom}}|}{|\boldsymbol{u}^{n+1}_{\text{rom}}|}$; $\epsilon^{n+1}_{p} = \frac{|p^{n+1}_{\text{rom}} - p^{n}_{\text{rom}}|}{|p^{n+1}_{\text{rom}}|}$; $\epsilon^{n+1}_{\boldsymbol{d}} = \frac{|\boldsymbol{d}^{n+1}_{\text{rom}} - \boldsymbol{d}^{n}_{\text{rom}}|}{|\boldsymbol{d}^{n+1}_{\text{rom}}|}$
  if $\epsilon^{n+1}_{\boldsymbol{u}}$ and ϵ^{n+1}_{p} and $\epsilon^{n+1}_{\boldsymbol{d}} \leq \text{Tol}_{\text{time}}$ **then**
    Stationary state achieved; break temporal loop
  end if
end for
Finalize case
Output if necessary

Out of experimentation it was found that ROM results that accurately represent the FOM are obtained when 99.999999% of the energy of the fluid is kept, amounting to 150 basis vectors, and 99.999999% of the energy of the solid base is taken, amounting to 36 basis vectors. ROM results are presented for this case.

DOI: 10.1002/nme

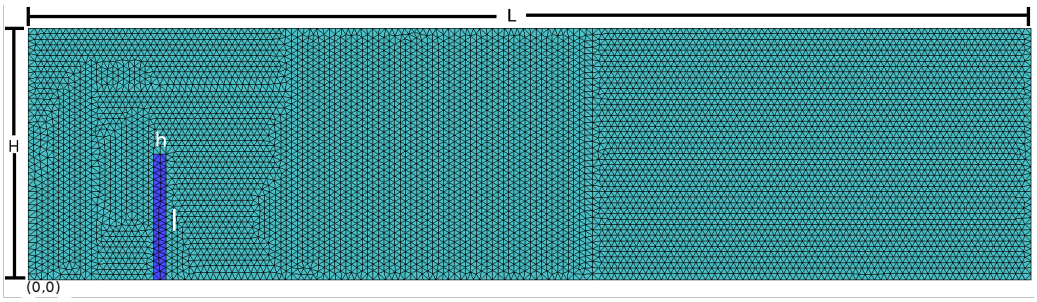


Figure 2. Geometry and mesh used for semi-stationary FSI-ROM case

Table II. Mesh parameters

	Fluid	Solid
Element type	Quadratic triangle	Quadratic triangle
Nodes per element	6	6
# of elements	14,308	78
# of nodes	29,057	201

Table III. Boundary conditions

Fluid	Solid
$x = 0$: $u_x = 1$, $u_y = 0$	$y = 0$: $d_x = d_y = 0$
$y = 0, H$: Free slip	Other boundaries: fluid tractions
$x = L$: Free	
Other boundaries: solid velocities	

Note that from Table II it can be calculated that for the fluid problem the amount of degrees of freedom (DOF) is 87,171 while for the solid it is 402. For the reduced problem we have 150 DOF for the fluid and 36 for the solid. This means that overall in terms of DOF we are achieving a reduction of 99.83% for the fluid and 91.05% for the solid, for a total reduction of 99.79%.

Figures 3 and 4 show contour plots for velocity and Figures 5 and 6 show contour plots for pressure for the final time of analysis, $t_{\mathrm{f}} = 10.0$ s. Figure 7 shows displacement contours in the solid.

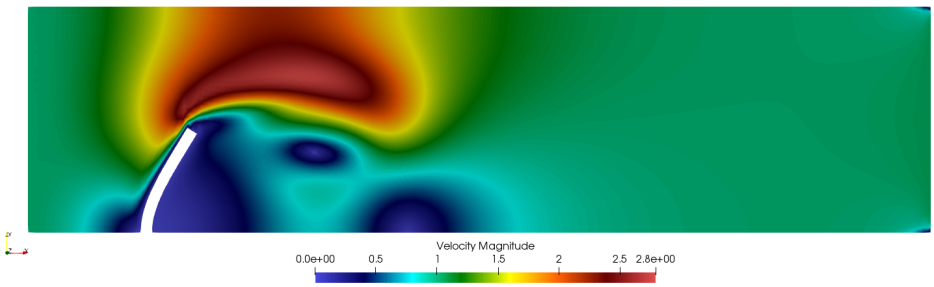


Figure 3. FOM - Velocity

Both solutions, FOM and ROM, are very similar for both the fluid and the solid. Differences can be better observed in the next plots, in which we also consider the dependency on the time interval of analysis with which snapshots to construct the ROM basis are taken. Results are shown for three particular cases, labeled ROM_A, ROM_B and ROM_C, leading to three different ROM basis. These are $\phi_{\mathrm{ROM\_A}}$, with a time interval $(0, 10)$, $\phi_{\mathrm{ROM\_B}}$, with a time interval $(0, 20)$, and $\phi_{\mathrm{ROM\_C}}$, with a time interval $(0, 40)$. In all cases, snapshots were taken each time step.

DOI: 10.1002/nme

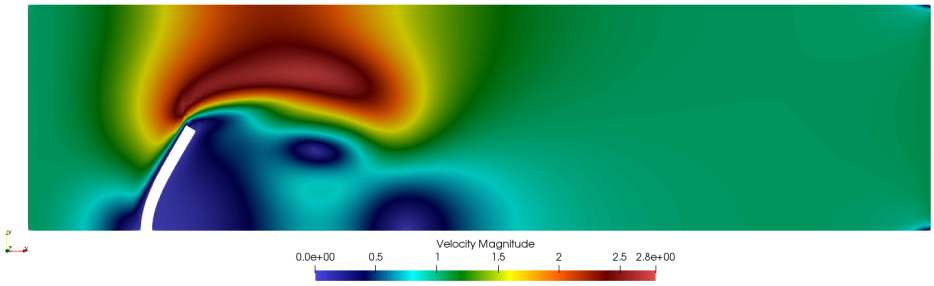


Figure 4. ROM - Velocity

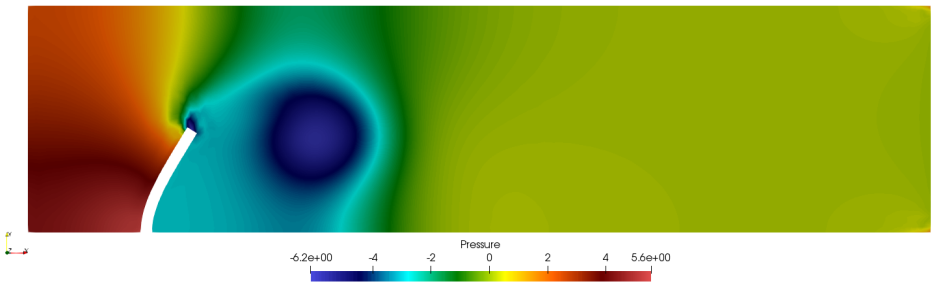


Figure 5. FOM - Pressure

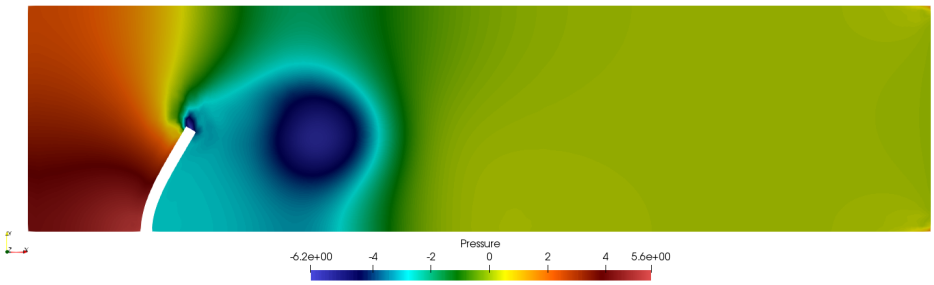


Figure 6. ROM - Pressure

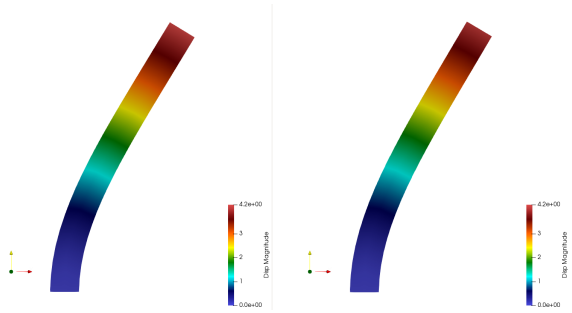
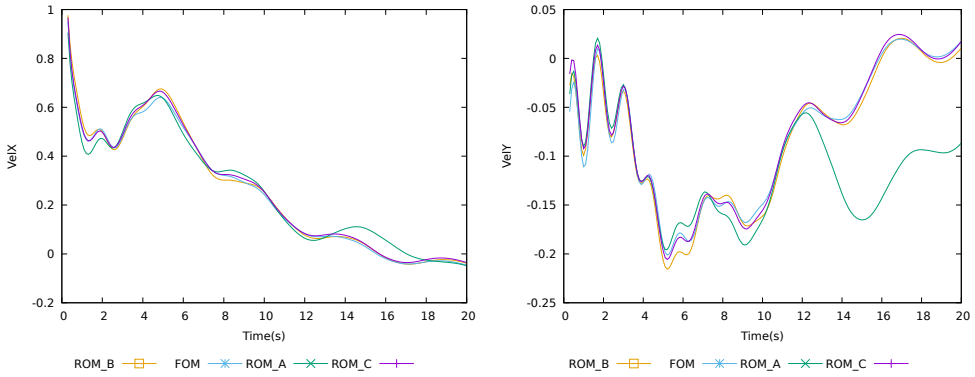


Figure 7. Displacement magnitude for the solid bar. Left: FOM, right: ROM

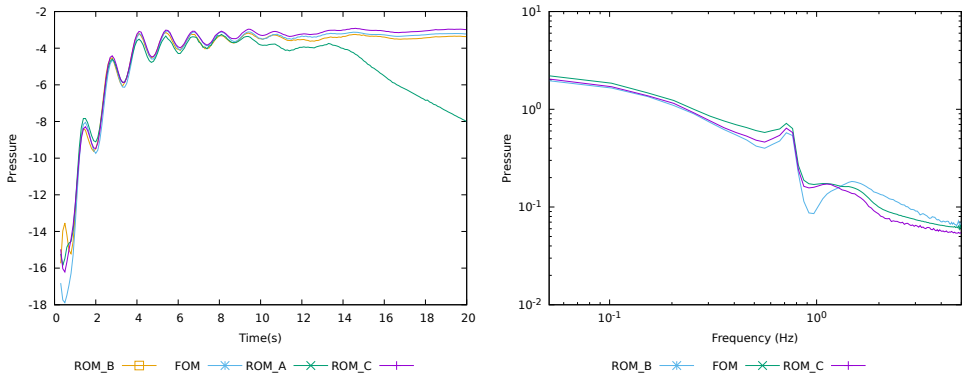
Figures 8 and 9 show the velocity components and pressure at a point in the fluid above the solid. Notice the importance of sampling the long stationary that develops after time $t = 12$. Even though cases ROM_B and ROM_C produce similar results, Figure 9b (the fast Fourier transform, FFT, of the pressure history) highlights that ROM_C yields a more stable and smoother solution.

DOI: 10.1002/nme



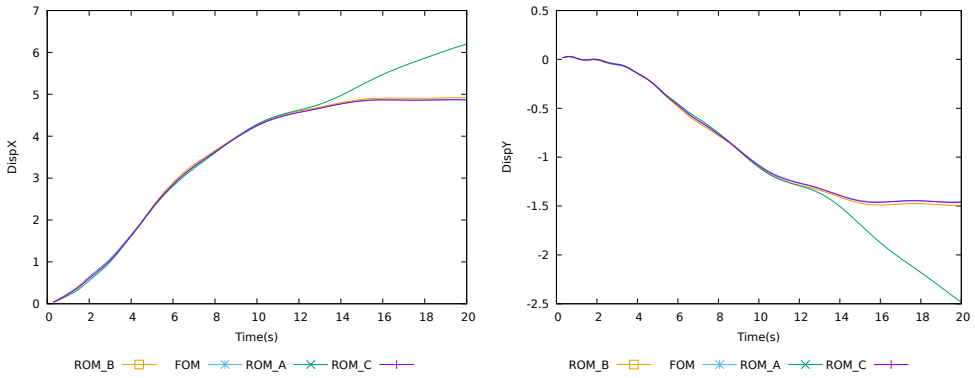
(a) Velocity in x axis (b) Velocity in y axis

Figure 8. x and y velocities above the plate



(a) Pressure (b) Pressure FFT

Figure 9. Pressure and its FFT around the plate

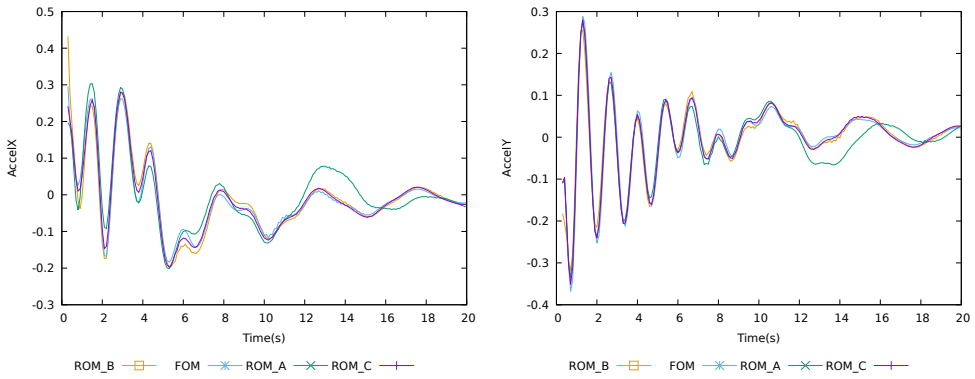


(a) Displacement in x axis (b) Displacement in y axis

Figure 10. Displacement of the tip of the plate

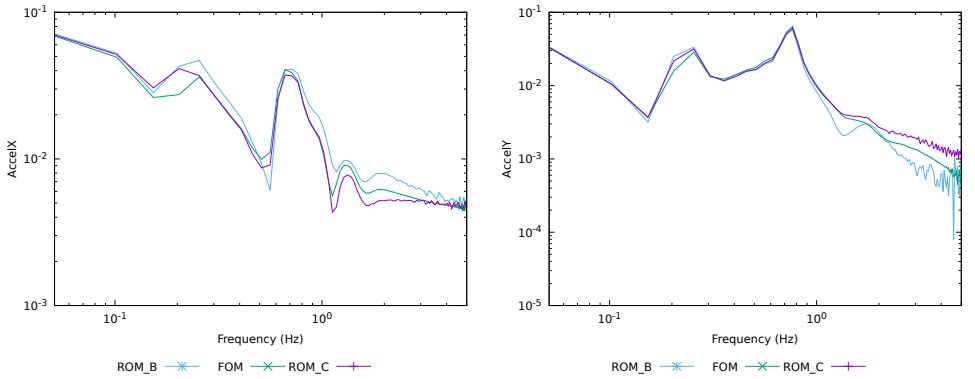
Figure 10 shows the displacement of the tip of the plate; once again it is important to notice that lack of sampling of the stationary part of the solution makes the ROM inaccurate only in this region.

DOI: 10.1002/nme



(a) Acceleration in x axis (b) Acceleration in y axis

Figure 11. Acceleration of the tip of the plate



(a) Acceleration FFT in x axis (b) Acceleration FFT in y axis

Figure 12. FFT of the acceleration of the tip of the plate

Figures 11 and 12 show the acceleration and its FFT at the tip of the plate. The analysis of the acceleration of the solid has been found to be critical, specially for FSI cases. It is this quantity that is really telling of the stability of the solid domain.

6.2. Flow around a cylinder with supported flag

The following example reproduces the benchmark presented in [38], where a fluid flows around a cylinder with a supported flag. The fluid flows from the left wall and the tractions of the fluid onto the solid initiate the flag motion. After a while this motion is significant enough to move the fluid around it, starting a feedback loop between fluid and solid. The test conditions are shown in Table IV.

Table IV. Physical parameters

	Fluid		Solid
ρ_{fl}	1,000.0	ρ_{s}	10,000.0
μ_{fl}	0.001	μ_{s}	$1.929 \cdot 10^6$
		λ_{s}	$7.714 \cdot 10^6$
Model	Newtonian		St.Venant-Kirchoff

DOI: 10.1002/nme

Figures 13 and 14 show the geometry and the finite element mesh used in this example. Note that the solid and fluid meshes are non-conforming, making the use of interpolation between sub-domains necessary as discussed in Section 4.2. The length of the fluid domain is $L = 2.5$, its height $H = 0.41$, and the radius of the cylinder is $R = 0.05$. The length of the bar is $l = 0.35$ and its thickness $h = 0.02$.

Figure 13. Geometry

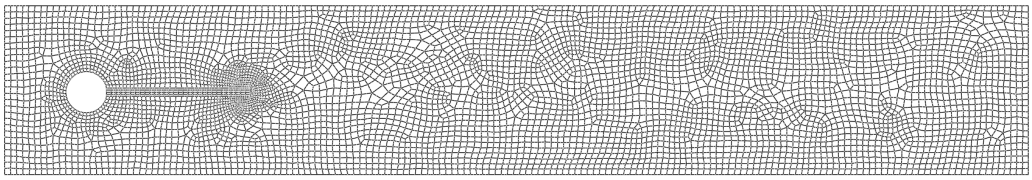


Figure 14. Non-conforming mesh

Figure 15 shows a zoom for the cylinder and bar.

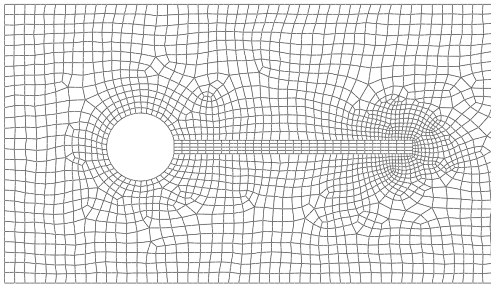


Figure 15. Non-conforming mesh - zoom

Tables V and VI show important mesh parameters and boundary conditions, respectively.

Table V. Mesh parameters

	Fluid	Solid
Element type	Quadratic quads	Quadratic quads
Nodes per element	9	9
# elements	5,531	500
# nodes	22,642	2,211

Out of experimentation it was found that ROM results that accurately represent the full order model are obtained when 99.999999% of the energy of the fluid is taken, amounting to 163 basis vectors, and 99.999999% of the energy of the solid is taken, amounting to 48 basis vectors. Results for a basis using 99.9999% of the energy are also shown in the following, in this case using 158 basis vectors for the fluid and 16 for the solid.

Notice that from Table V it can be calculated that for the fluid problem the amount of DOF is 67,926 while for the solid it is 4,422. For the reduced problem we have 163 DOF for the fluid and 48 for

DOI: 10.1002/nme

Table VI. Boundary conditions

	Fluid	Solid
$x = 0$:	$u_x = 35.693\, y(0.41 - y)$, $u_y = 0.0$	
$y = 0, H$:	Free slip	
$x = L$:	Free (zero traction)	
Cylinder boundary:	No slip	
Flag boundary:	Solid velocity	Fluid tractions
Flag-cylinder union:		$d_x = d_y = 0$

the solid. This means that overall in terms of DOF we are achieving a reduction of 99.76% for the fluid and 98.91% for the solid, for a total reduction of 99.71%.

The following collection of figures show contours for velocity and pressure, for both the reduced order problem and the full order problem at the last time of the simulation, $t = 1.2$. After this, graphs of significant quantities are compared for both the reduced and full order problems.

Figures 16 and 17 show velocity contours, Figures 18 and 19 show pressure contours, and Figure 20 shows the strain contours for the solid domain for both problems. In all cases, solutions are very similar.

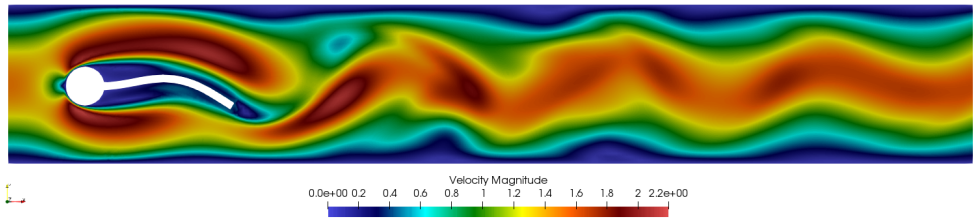


Figure 16. FOM - Velocity magnitude

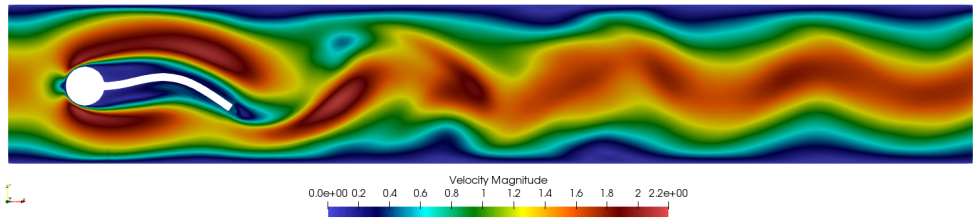


Figure 17. ROM - Velocity magnitude

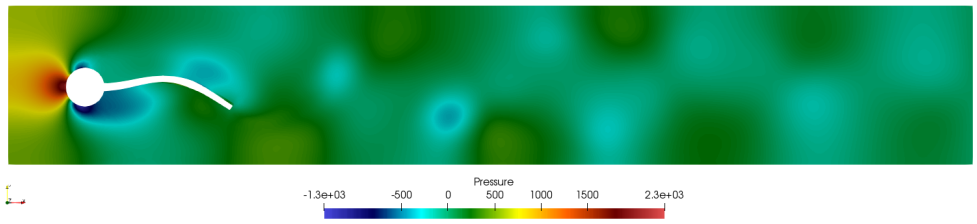


Figure 18. FOM - Pressure

DOI: 10.1002/nme

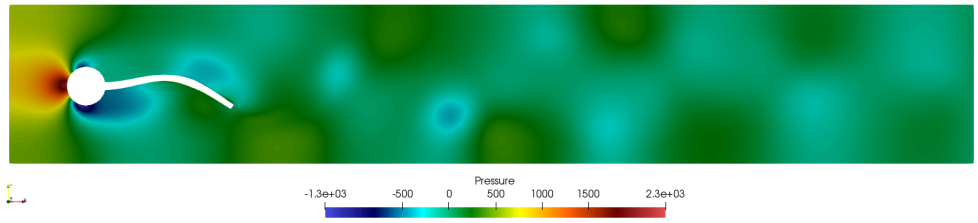


Figure 19. ROM - Pressure

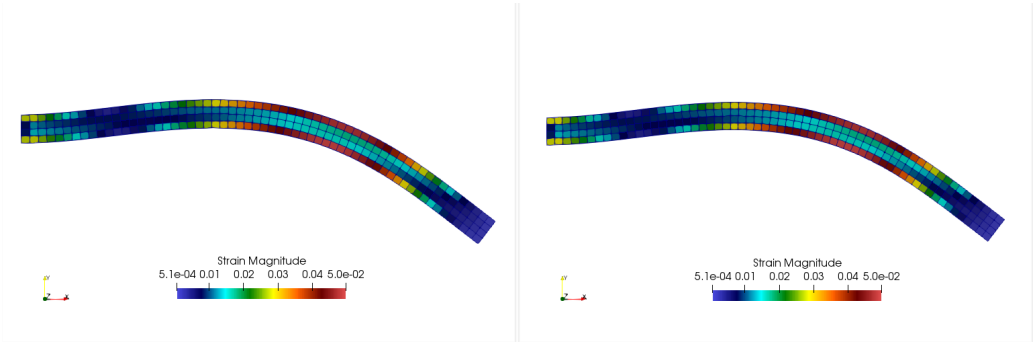
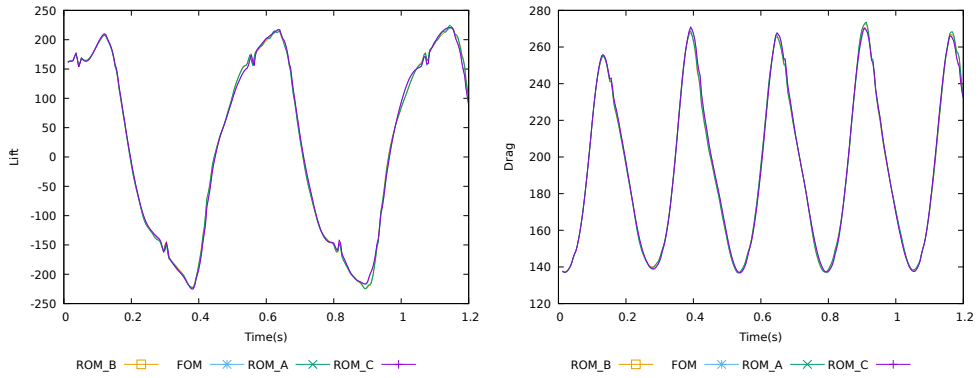


Figure 20. Strain magnitude for the solid bar. Left: FOM, right: ROM

Out of the many results that can be shown, it is considered valuable to see the dependency of the ROM result on the energy percentage used. Results are shown for three particular cases, labeled ROM_A, ROM_B and ROM_C, which correspond to $\eta = 99.9999\%$ the first two and $\eta = 99.999999\%$ the last one. The bases for all three cases were obtained sampling every time-step the FOM solution to collect the snapshots. While cases ROM_A and ROM_B share the same basis and same energy percentage, the difference between them is the stabilization constant for the incompressibility term of the Navier-Stokes equation (see Eq. (10)). In this example we explore the effect of a slight variation in constant c_3. For case ROM_A we have used $c_3 = 1.5$, while for case ROM_B we have taken $c_3 = 2.0$.

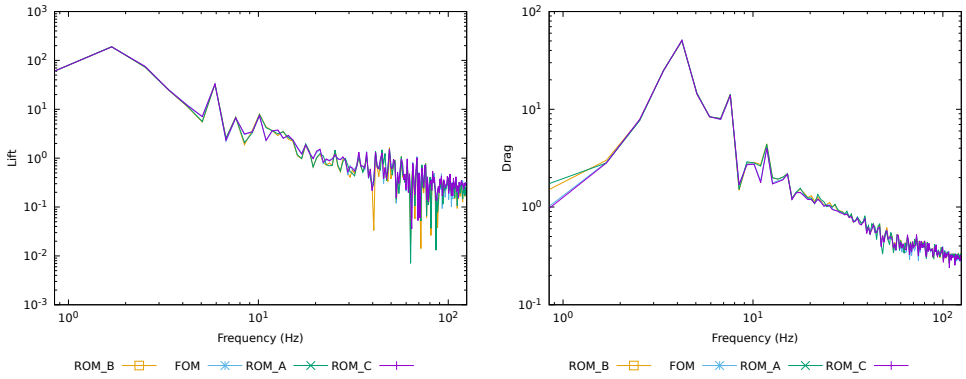


(a) Lift (b) Drag

Figure 21. Lift and drag around the flag

Figures 21 and 22 show the drag and lift around the geometry of the flag caused by the fluid. Unlike the example shown in Section 6.1, this test case is much more complex and requires much more computational time as well as a richer basis to produce meaningful results.

DOI: 10.1002/nme



(a) Lift FFT (b) Drag FFT

Figure 22. FFT of the lift and drag around the flag

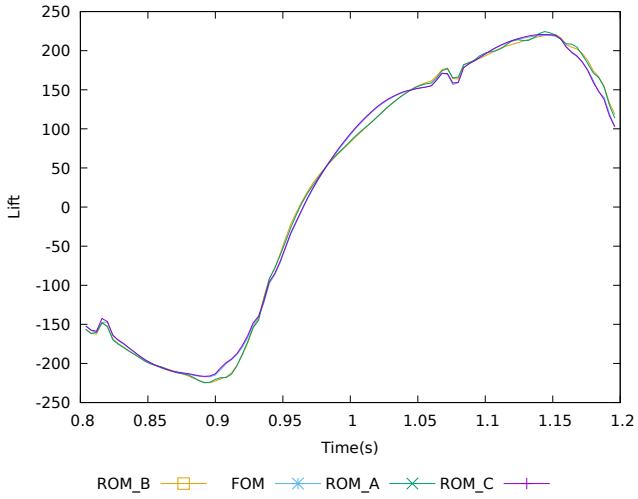
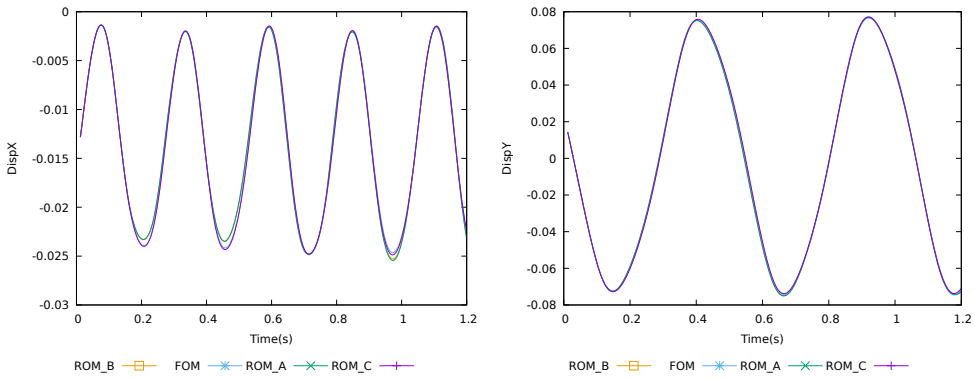


Figure 23. Zoom for lift

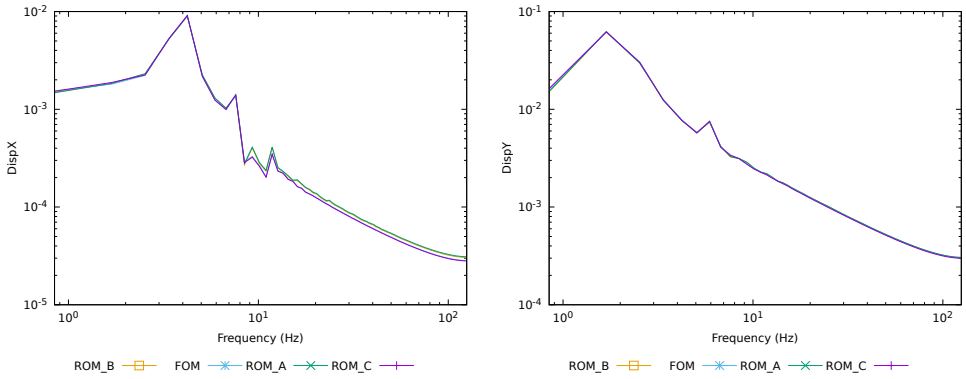
Figure 23 shows a zoom for the lift around the geometry of the flag.



(a) Displacement in x axis (b) Displacement in y axis

Figure 24. Displacement at the tip of the flag

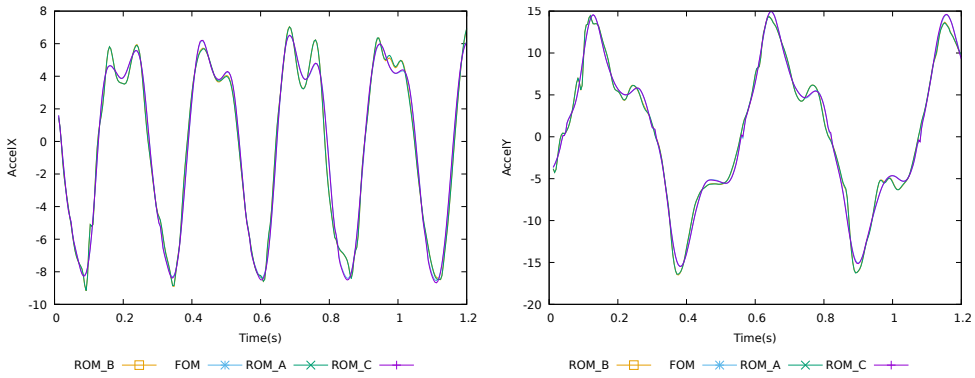
DOI: 10.1002/nme



(a) Displacement FFT in x axis (b) Displacement FFT in y axis

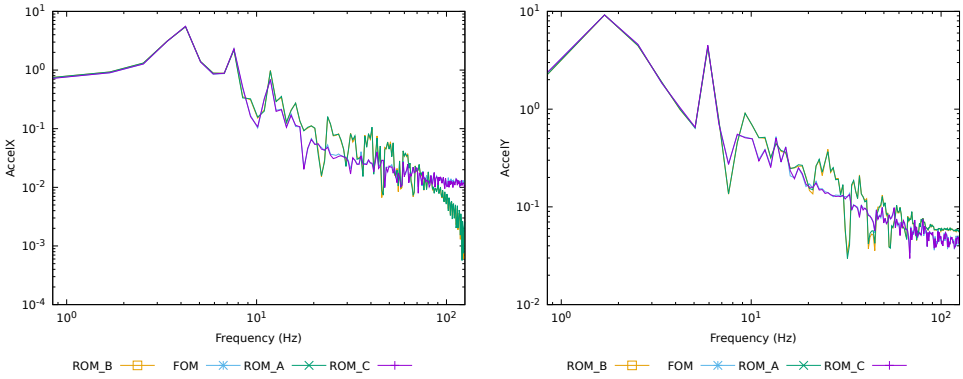
Figure 25. FFT of the displacement at the tip of the flag

Figures 24 and 25 show the displacement and its Fourier's transform at the tip of the flag. All results reproduce accurately the FOM.



(a) Acceleration in x axis (b) Acceleration in y axis

Figure 26. Acceleration at the tip of the flag



(a) Acceleration FFT in x axis (b) Acceleration FFT in y axis

Figure 27. FFT of the acceleration at the tip of the flag

DOI: 10.1002/nme

Figures 26 and 27 show the acceleration and its Fourier's transform at the tip of the flag. In this case the analysis of the acceleration of the flag is not only interesting but crucial. It can be seen that taking a higher stabilization constant for the incompressibility term changes the accuracy with which a ROM case reproduces the FOM. The acceleration is thus a variable very sensitive to the dimension of the ROM space.

Table VII shows the total times and speedups for all cases shown. It has to be remarked that we have not used any hyper-reduction strategy, and therefore these speedups could be improved. Note that the number of DOF in the fluid for the ROM is just 0.24% of that of the FOM for the richest ROM (163 DOF for the ROM vs. 67,926 DOF for the FOM). An ideal implementation would yield a reduction in the computational time solely determined by the number of DOF.

Table VII. Time and speedup for fluid and solid domains

		FOM	ROM_A	ROM_B	ROM_C
Fluid	Time(min)	88.53	16.21	19.33	18.94
	Speedup		81.69%	78.16%	78.6%
Solid	Time(min)	0.79	0.54	0.63	0.684
	Speedup		31.65%	20.25%	13.42%
Total	Time(min)	89.32	16.75	19.96	19.624
	Speedup		81.24%	77.65%	78.3%

7. CONCLUSIONS

In this paper we have proposed a ROM model for FSI problems based on the VMS framework. The first conclusion refers to the use of ROM in FSI. In comparison to the work done in [33], where the cases shown could be solved with a basis energy in the range of 80% to 95%, FSI problems seem to be much more sensitive to the amount of energy in the basis necessary to achieve a solution sufficiently close to that of the FOM. It was found that even for the simplest of the cases shown, the problem would not produce any valuable solution with an energy percentage under 99.0%. It is possible that this hints to the importance of the high frequencies of the spectrum in the solution of a FSI problem. This remains to be studied further and it is an interesting topic for future work. Nevertheless, in spite of the stringent requirements in terms of retained energy by the ROM, the reduction of the number of DOF with respect to the FOM is still very remarkable.

Once the possibility of reducing drastically the number of DOF has been shown, we have not pursued an efficient implementation of the highly nonlinear problems involved in both the solid and the fluid domains. In particular, we have not implemented any hyper-reduction strategy, which should be used on top of the ROM we have proposed. We are currently working on a hyper-reduction method based on the existence of a finite element mesh for the ROM.

Partitioned FSI problems have a series of restrictions, such as a maximum time step and added diffusivity, that must be met so as to minimize the effect of instabilities like the added mass effect. This in turn is also a restriction on the ROM, making FSI-ROM cases very dependent on how often a snapshot is taken to be able to capture enough of the physics of the problem while keeping instabilities out of the sampling. A richer basis consists of more basis vectors that contain higher frequencies, and in turn, produce a better approximation to the FOM problem. Again, correlating with the above, this hints to the dependency on the high frequency low energy modes at the end of the spectrum of the basis.

The VMS-ROM formulation we have proposed for the fluid domain has been found to be accurate and efficient. From the theoretical point of view, it has the interesting feature that the ROM and the FOM problems are solved exactly with the same formulation, only changing the spaces where

DOI: 10.1002/nme

the unknowns and the test functions belong. From the practical point of view, we have observed that ROM problems with a higher stabilization constant for the incompressibility produced more accurate results that their counterparts with a lower one. We have found convenient to use values of c_3 in Eq. (10) slightly higher in the ROM than in the FOM.

ACKNOWLEDGEMENTS

Alexis Tello wants to acknowledge the doctoral scholarship received from the Colombian Government–Colciencias. Ramon Codina acknowledges the support received from the ICREA Acadèmia Research Program of the Catalan Government. Joan Baiges acknowledges the support of the Spanish Government through the Ramón y Cajal grant RYC-2015-17367. This work is partially funded through the ELASTIC-FLOW project, Ref. DPI2015-67857-R of the Spanish Government.

REFERENCES

1. Muzaffer Akbay, Nicholas Nobles, Victor Zordan, and Tamar Shinar. An extended partitioned method for conservative solid-fluid coupling. *ACM Transactions on Graphics*, 37(4):1–12, 7 2018.
2. Santiago Badia, Fabio Nobile, and Christian Vergara. Fluidstructure partitioned procedures based on Robin transmission conditions. *Journal of Computational Physics*, 227(14):7027–7051, 2008.
3. Joan Baiges. *The Fixed-Mesh ALE method applied to multiphysics problems using stabilized formulations*. PhD thesis, Universitat Politècnica de Catalunya, 2011.
4. Joan Baiges, Ramon Codina, and Sergio Idelsohn. A domain decomposition strategy for reduced order models. Application to the incompressible NavierStokes equations. *Computer Methods in Applied Mechanics and Engineering*, 267:23–42, 12 2013.
5. Joan Baiges, Ramon Codina, and Sergio Idelsohn. Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 72(12):1219–1243, 8 2013.
6. Joan Baiges, Ramon Codina, and Sergio Idelsohn. Reduced-order subscales for POD models. *Computer Methods in Applied Mechanics and Engineering*, 291:173–196, 7 2015.
7. Joan Baiges, Ramon Codina, Arnau Pont, and Ernesto Castillo. An adaptive Fixed-Mesh ALE method for free surface flows. *Computer Methods in Applied Mechanics and Engineering*, 313:159–188, 1 2017.
8. Francesco Ballarin and Gianluigi Rozza. POD-Galerkin monolithic reduced order models for parametrized fluid-structure interaction problems. *International Journal for Numerical Methods in Fluids*, 82(12):1010–1034, 12 2016.
9. Yuri Bazilevs, V. M. Calo, T. J R Hughes, and Y. Zhang. Isogeometric fluid-structure interaction: theory, algorithms, and computations. *Computational Mechanics*, 43(1):3–37, 12 2008.
10. Yuri Bazilevs, V. M. Calo, Y. Zhang, and T. J R Hughes. Isogeometric Fluidstructure Interaction Analysis with Applications to Arterial Blood Flow. *Computational Mechanics*, 38(4-5):310–322, 9 2006.
11. Ted Belytschko, Wing Kam Liu, Brian Moran, and Khalil Elkhodary. Nonlinear finite elements for continua and structures. page 830. Wiley, 2nd edition, 2014.
12. Sylvie Bordère and J.-P. Caltagirone. A unifying model for fluid flow and elastic solid deformation: A novel approach for fluidstructure interaction. *Journal of Fluids and Structures*, 51(1):344–353, 11 2014.
13. Giorgio Chiandussi, Gabriel Bugeda, and Eugenio Oñate. A simple method for automatic update of finite element meshes. *Communications in Numerical Methods in Engineering*, 16(1):1–19, 1 2000.
14. Ramon Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190(13-14):1579–1599, 12 2000.
15. Ramon Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190(20-21):2681–2706, 2 2001.
16. Ramon Codina. Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *Computer Methods in Applied Mechanics and Engineering*, 191(39-40):4295–4321, 8 2002.
17. Ramon Codina, Santiago Badia, Joan Baiges, and Javier Principe. Variational Multiscale Methods in Computational Fluid Dynamics. In *Encyclopedia of Computational Mechanics Second Edition*, pages 1–28. John Wiley & Sons, Ltd, Chichester, UK, 2018.
18. Claudia Maria Colciago and Simone Deparis. Reduced order models for fluid-structure interaction problems with applications in haemodynamics. 1 2018.
19. Jean Donea, Antonio Huerta, Jean P. Ponthot, and Antonio Rodriguez-Ferran. Arbitrary Lagrangian-Eulerian Methods. In *Encyclopedia of Computational Mechanics*, pages 1–25. John Wiley & Sons, Ltd, Chichester, UK, 11 2004.
20. Richard Everson and Lawrence Sirovich. KarhunenLoève procedure for gappy data. *Journal of the Optical Society of America A*, 12(8):1657, 8 1995.

DOI: 10.1002/nme

21. Jiang Fan, Huming Liao, Renjie Ke, Erdem Kucukal, Umut A. Gurkan, Xiuli Shen, Jian Lu, and Bo Li. A monolithic Lagrangian meshfree scheme for FluidStructure Interaction problems within the OTM framework. *Computer Methods in Applied Mechanics and Engineering*, 337:198–219, 8 2018.
22. Charbel Farhat and Vinod K. Lakshminarayan. An ALE formulation of embedded boundary methods for tracking boundary layers in turbulent fluid-structure interaction problems. *Journal of Computational Physics*, 263:53–70, 2014.
23. Charbel Farhat, Kristoffer G. van der Zee, and Philippe Geuzaine. Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 195(17-18):1973–2001, 2006.
24. Christiane Förster, Wolfgang A Wall, and Ekkehard Ramm. Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 196(7):1278–1293, 1 2007.
25. Swetlana Giere, Traian Iliescu, Volker John, and David Wells. SUPG reduced order models for convection-dominated convection-diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 289:454–474, 2015.
26. Roland Glowinski, Steffen Basting, and Annalisa Quaini. Extended ALE Method for fluidstructure interaction problems with large structural displacements. *Journal of Computational Physics*, 331:312–336, 2017.
27. Gene Hou, Jin Wang, and Anita Layton. Numerical methods for fluid-structure interaction - A review. *Communications in Computational Physics*, 12(2):337–377, 2012.
28. Guillaume Houzeaux and Ramon Codina. Transmission conditions with constraints in finite element domain decomposition methods for flow problems. *Communications in Numerical Methods in Engineering*, 17(3):179–190, 2 2001.
29. Thomas J.R. Hughes, Gonzalo R. Feijóo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale methoda paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166(1-2):3–24, 1998.
30. Ulrich Küttler and Wolfgang A. Wall. Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43(1):61–72, 2008.
31. Ulrich Langer and Huidong Yang. Partitioned solution algorithms for fluidstructure interaction problems with hyperelastic models. *Journal of Computational and Applied Mathematics*, 276:47–61, 3 2015.
32. Patrick Le Tallec. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190(24-25):3039–3067, 2001.
33. Ricardo Reyes, Ramon Codina, Joan Baiges, and Sergio Idelsohn. Reduced order models for thermally coupled low Mach flows. *Advanced Modeling and Simulation in Engineering Sciences*, 5(1):28, 12 2018.
34. Roger A Sauer and Tobias Luginsland. A monolithic fluidstructure interaction formulation for solid and liquid membranes including free-surface contact. *Computer Methods in Applied Mechanics and Engineering*, 341:1–31, 11 2018.
35. Wil Schilders. Introduction to Model Order Reduction. In *Model Order Reduction: Theory, Research Aspects and Applications. Mathematics in Industry (The European Consortium for Mathematics in Industry)*, volume 13, pages 3–32. Springer, Berlin, Heidelberg, 2008.
36. Lawrence Sirovich and Michael Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519, 3 1987.
37. Ali Thari, Vito Pasquariello, Niels Aage, and Stefan Hickel. Adaptive Reduced-Order Modeling for Non-Linear Fluid-Structure Interaction. pages 1–25, 2 2017.
38. Stefan Turek and Jaroslav Hron. Proposal for Numerical Benchmarking of Fluid-Structure Interaction between an Elastic Object and Laminar Incompressible Flow. In *Fluid-Structure Interaction*, volume 53, pages 371–385. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
39. Dunhui Xiao, Pan Yang, Fangxin Fang, Jiansheng Xiang, Chris C. Pain, and Ionel M. Navon. Non-intrusive reduced order modelling of fluidstructure interactions. *Computer Methods in Applied Mechanics and Engineering*, 303:35–54, 5 2016.

DOI: 10.1002/nme